NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

FUZZY CLUSTERING MEANS ALGORITHM FOR TRACK FUSION IN U.S. COAST GUARD VESSEL TRAFFIC SERVICE SYSTEMS

by

Eugenio S. Anzano

June 1999

Thesis Advisor: Murali Tummala

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188.

1. AGENCY USE ONLY *(Leave blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	June 1999	Master's Thesis

4. TITLE AND SUBTITLE
FUZZY CLUSTERING MEANS ALGORITHM FOR TRACK FUSION IN U.S. COAST GUARD VESSEL TRAFFIC SERVICE SYSTEMS

5. FUNDING NUMBERS

6. AUTHOR(S)
Eugenio S. Anzano

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES
The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release, distribution is unlimited.

12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*
This thesis presents a fuzzy association based data fusion algorithm for U.S. Coast Guard Vessel Traffic Service (VTS) systems to reduce the number of redundant target tracks displayed to vessel traffic operators. The proposed algorithm uses the Fuzzy Clustering Means (FCM) algorithm to reduce the number of target tracks and associate duplicate tracks by determining the degree of membership for each target track. The algorithm uses current sensor data and the known sensor resolutions for measurement-to-measurement association and the selection of the most accurate sensor for tracking fused targets. Actual vessel traffic data collected from U.S. Coast Guard VTS systems are used for simulation and analysis of the algorithm. The results exhibit successful fusion of correlated tracks and selection of the most accurate sensor resulting in a reduced number of tracks displayed to the VTS operator.

14. SUBJECT TERMS	15. NUMBER OF PAGES
Vessel Traffic Services, VTS, Redundant Observations, Data Fusion, Fuzzy Logic, Fuzzy Clustering Means	93
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UL

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18, 298-102

Approved for public release; distribution is unlimited.

FUZZY CLUSTERING MEANS ALGORITHM FOR TRACK FUSION IN U.S. COAST GUARD VESSEL TRAFFIC SERVICE SYSTEMS

Eugenio S. Anzano
Lieutenant Junior Grade, United States Coast Guard
B.S.E.E., U.S. Coast Guard Academy, 1995

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 1999

Author: Eugenio S. Anzano

Eugenio S. Anzano

Approved by: Murali Tummala

Murali Tummala, Thesis Advisor

Roberto Cristi, Second Reader

Jeffrey B. Knorr, Chairman
Department of Electrical and Computer Engineering

ABSTRACT

This thesis presents a fuzzy association based data fusion algorithm for U.S. Coast Guard Vessel Traffic Service (VTS) systems to reduce the number of redundant target tracks displayed to vessel traffic operators. The proposed algorithm uses the Fuzzy Clustering Means (FCM) algorithm to reduce the number of target tracks and associate duplicate tracks by determining the degree of membership for each target track. The algorithm uses current sensor data and the known sensor resolutions for measurement-to-measurement association and the selection of the most accurate sensor for tracking fused targets. Actual vessel traffic data collected from U.S. Coast Guard VTS systems are used for simulation and analysis of the algorithm. The results exhibit successful fusion of correlated tracks and selection of the most accurate sensor resulting in a reduced number of tracks displayed to the VTS operator.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS, ACRONYMNS, AND/OR ABBREVIATIONS

VTS	Vessel Traffic Service
USCG	United States Coast Guard
VTC	Vessel Traffic Center
GPS	Global Positioning System
ADS	Automated Dependent Surveillance
SR	Standard Route
CCTV	Closed CircuitTelevision
Tbdm	Track Data Base Manager
AOR	Aarea of Responsibility
RSS	Remote Site Subsytem
VTCS	Vessel Traffic Control Subsystem
NM	Nautical Mile
DGPS	Differential Global Positioning System
EP	Estimated Position
INRI	Inter-National Research Institute
UTC	Universal Time Code
MSMT	Multisensor/Multitarget
IFF	Interrogation Friend or Foe
χ	Characteristic Function
μ	Fuzzy Membership Function
α	Association Threshold Value
FCM	Fuzzy Clustering Means

ACKNOWLEDGMENT

The author would like to thank and acknowledge Prof. Murali Tummala, Prof. Roberto Cristi, and Maj. Ashraf Aziz for their exceptional help, guidance, and patience during the completion of this research. The author would also like to thank the personnel of VTS San Francisco and VTS Puget Sound for their technical knowledge and assistance in collecting data and sensor information at their facilities.

I. INTRODUCTION

The United States Coast Guard (USCG) operates and maintains the Vessel Traffic Services (VTS) System that oversees and manages maritime vessel traffic in the harbors and waterways of the United States. The VTS is a communications and surveillance network designed to coordinate the safe and efficient transit of vessels in an effort to prevent accidents and the associated loss of life and damage to property and the environment. There are nine Vessel Traffic Centers (VTC) located throughout the country from Staten Island, New York, to Valdez, Alaska.

A common problem found in VTS systems is overlapping radar coverage of a waterway, which gives rise to duplicate target readings of a single vessel or object. When two or more radars sweep the same geographic area as shown in Figure 1, duplicate target/tracks appear on the VTS operator's display. This is caused by numerous, sometimes uncontrollable factors such as different target viewing angles, measurement geometries, and sensor accuracy and resolution. VTS operators must manually decide which reading is the actual vessel, which may decrease the efficiency and ability of the operator to coordinate traffic. The complexity of this problem increases with multiple vessels transiting in close proximity, causing the readings of one vessel to be indistinguishable from the other. A solution to this problem is to correlate these duplicate radar tracks using fuzzy association based data fusion algorithms.

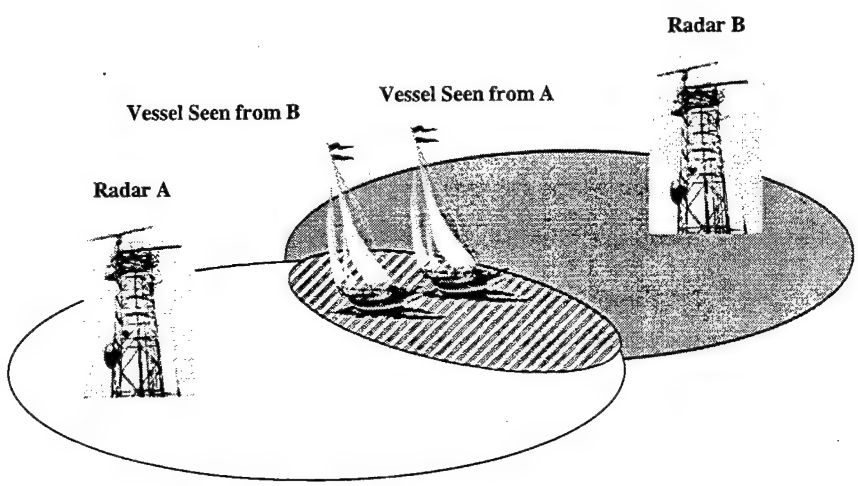


Figure 1.1. Overlapping Radar Coverage

Data fusion algorithms help determine which track the vessel belongs to and automatically associate the duplicate tracks to one "unique" track. This frees the VTS operator of the manual task of associating ambiguous tracks, which will overall improve the operator's efficiency in managing vessel traffic.

A. GOAL OF THE THESIS

The main thrust of this study is to minimize duplicate target readings and fuse them into individual tracks for each vessel using fuzzy association based data fusion techniques. The data fusion functionality is integrated into VTS software as shown in Figure 1.2. This thesis will focus on an algorithm, proposed by Aziz [1,2], that optimizes the degree of membership for each target track and selects the "best" sensor for reporting the vessel(s) in the area of overlapping radar coverage. The similarity measures between the reports from different sensors and the corresponding resolutions of each sensor provide the fuzzy membership functions for the decision process, which determines if duplicate tracks are from the same target. Unlike previously reported fuzzy logic data association algorithms [3,4,5], the proposed algorithm performs data association based on the data received from the radars and the known accuracies of the sensors. The main advantages of this algorithm are its simplicity and efficiency in its application to dense target environments with multiple sensors and sensor attributes. Also, this algorithm is computationally less expensive than conventional fuzzy logic data association techniques.

Actual traffic data from USCG VTS systems are used to test the fusion algorithm. Vessel traffic data from VTS Puget Sound, previously collected in 1996 [3,4], and data from VTS San Francisco, collected in 1999, are utilized to test the algorithm.

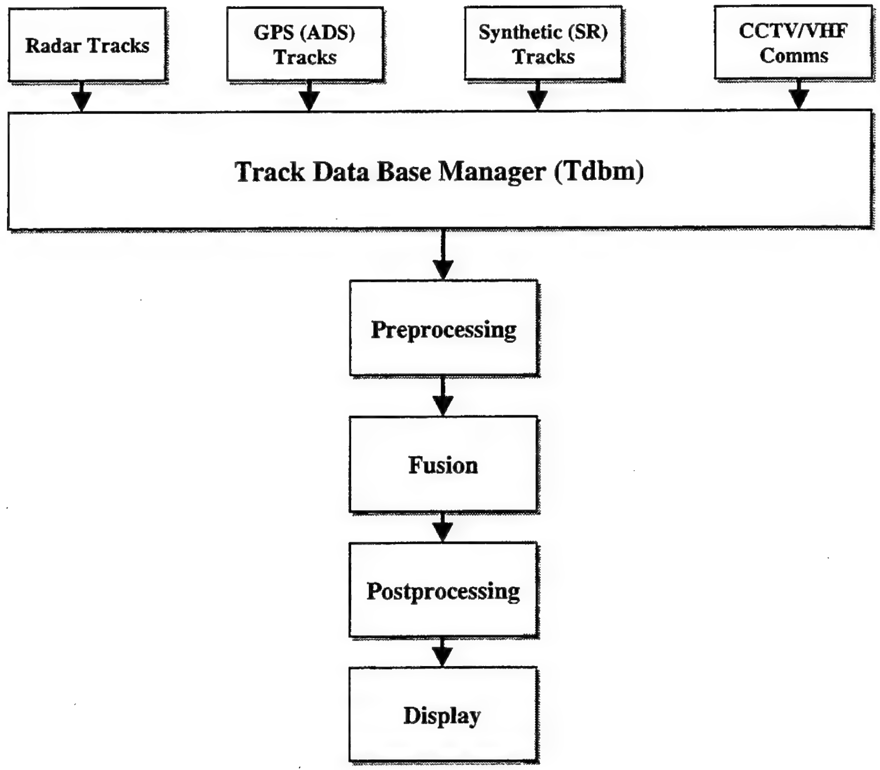


Figure 1.2. Overview of Fusion Algorithm

B. THESIS OUTLINE

The remainder of the thesis is organized as follows. A description of the VTS environment is provided in Chapter II. The overall VTS system and the method of data collection are discussed in this chapter. A discussion of multisensor data fusion and fuzzy association are found in Chapters III and IV, respectively. Chapter V contains a detailed explanation of the fusion algorithm used in this study. Chapter VI describes the simulation process and depicts the test results. Chapter VII ends the thesis with conclusions and suggestions for further development. Track status codes for VTS San Francisco data are listed in Appendix A. Appendix B contains the data capture algorithm for formatting traffic data, and Appendix C contains the MATLAB® code developed to implement the fusion algorithm.

II. VTS ENVIRONMENT

VTS systems monitor waterways and harbors with remote sites that broadcast vessel traffic information to the VTC by way of transmission line or microwave links. The number of remote sites depends upon the geographic size and traffic density of the navigable area. For example, VTS San Francisco consists of four remote sites that monitor the San Francisco Bay area whereas VTS Puget Sound has thirteen remote sites that monitor Puget Sound and most of the Pacific Northwest passage. In some cases, the area of responsibility (AOR) of one remote site may extend into another site's AOR, resulting in redundant tracks displayed to the VTS operator. This chapter presents an overview of the VTS system, the unique features of the systems in Puget Sound and San Francisco Bay, and the methods of data collection at each VTS.

A. SYSTEM OVERVIEW

Each VTS system can be broken down into two subsystems: Remote Site Subsytems (RSS) and a centralized Vessel Traffic Control Subsystem (VTCS). As shown in Figure 2.1, each RSS consists of a radar unit, radar data processor, remote site processor, video cameras, video compressors, and a VHF radio system. The models and types of radar unit vary for each remote site, depending upon the range and power required to track vessels. Each remote site has two radar units for backup purposes in the event one unit fails. The radar data processor processes raw radar data to generate radar tracks that are sent to the VTCS. Sensor level fusion and track updates are also conducted at the radar data processor. The remote site processor provides an interface for the operator at the VTCS to control and monitor the RSS. This processor routes control signals from the VTCS to the RSS to change the position or zoom feature of the video camera or to change the power setting of the radar. The video cameras are typically black and white cameras with simple pan and zoom capabilities. The VHF radio system of the RSS is used by the VTS to broadcast vessel traffic information and to communicate with vessel pilots. Communication links between the RSS and the VTCS are either T1 transmission lines or microwave links.

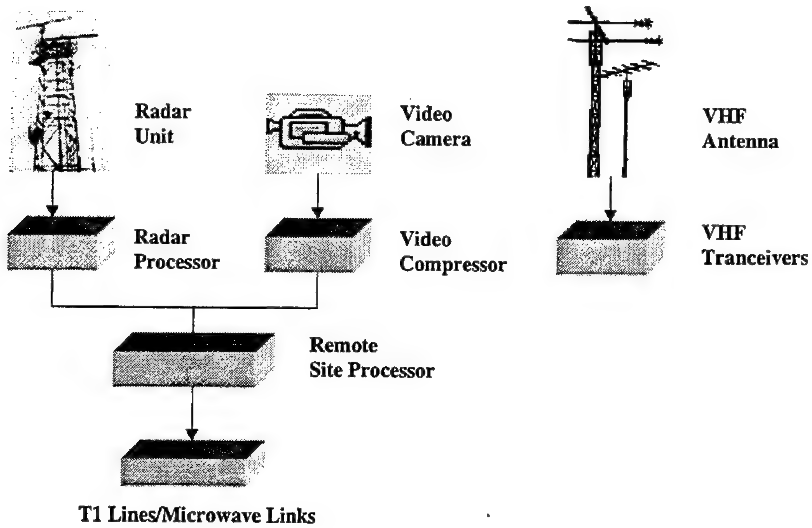


Figure 2.1. Remote Site Subsystem (RSS)

The VTCS is located at the VTS and consists of sensor data processors, audio/video routing systems, database processors, and operator display processors, as shown in Figure 2.2. The sensor data processor receives all incoming sensor data from each RSS. Audio and video signals sent from the cameras and VHF radio system are processed through the A/V routing system for presentation to the VTS operator and/or recording on audio and video tape. The database processor records traffic information from the sensor data processor and archives the information for historical record. The operator display processor allows the VTS operator to setup and control remote sites and to select specific data, or target tracks for display. Vessel traffic information, video images, and radio traffic are available to the VTS operator at the display console.

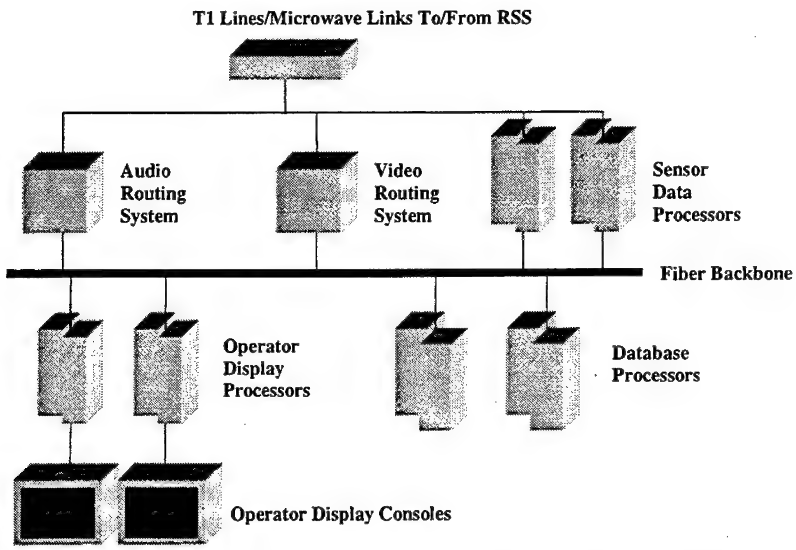


Figure 2.2. Vessel Traffic Control Subsystem (VTCS)

Each of the nine USCG VTS systems is structured with the remote and centralized subsystems. Features unique to each VTS are the number of remote sites and the types of sensors used to monitor vessel traffic. Two systems researched for this study are VTS Puget Sound and VTS San Francisco.

1. VTS Puget Sound

VTS Puget Sound monitors approximately 2,900 square miles of navigable waterways from the ocean shores of Washington State to as far inland as Commencement Bay. It consists of thirteen radar sites, fourteen communication sites, and three closed circuit television (CCTV) camera sites. Figure 2.2 shows VTS Puget Sound's area of responsibility.

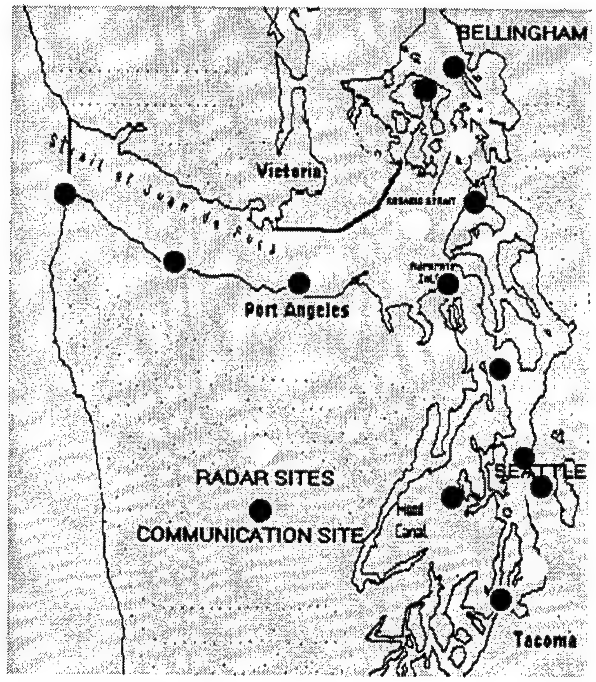


Figure 2.3. VTS Puget Sound Area of Responsibility [6]

All thirteen radar sites at VTS Puget Sound use the AIL FPS-109 search radar, and the locations of their sites are listed in Table 2.1. The known range and bearing resolutions for the AIL radar are 0.375% and ±0.35 degrees, respectively. Range resolution in yards is determined by multiplying the range resolution in percentage by the distance to the target. For example, a vessel 12 NM away produces a range resolution of 0.045 NM or 91.14 yards, i.e., the vessel has a range error of 91.14 yards. The farther away the vessel is from the radar, the less accurate the track is at the operator's display.

Radar Site	Site Location
Cape Flattery	48° 23' 14" N 124° 42' 51" W
Pearson Creek	48° 15' 41" N 124° 14' 16" W
Port Angeles	48° 08' 24.5" N 123° 24' 37.5" W
Shannon Point	48° 30' 32.5" N 122° 40' 56" W
Village Point	48° 43' 14" N 122° 42' 46" W
Smith Island	48° 19' 13" N 122° 50' 29" W
Whidbey Island	48° 19' 00" N 122° 41' 53" W
Point Wilson	48° 08' 37" N 122° 45' 14" W
Point No Point	47° 54' 43" N 122° 31' 34" W
West Point	47° 39' 44" N 122° 25' 58" W
Pier 36	47° 35' 24" N 122° 20' 58" W
Ruston	47° 18' 09" N 122° 20' 58" W
Pt Robinson	47° 23' 16" N 122° 22' 29" W

Table 2.1. VTS Puget Sound Radar Sites

2. VTS San Francisco

VTS San Francisco covers 133 miles of waterway covering the Offshore Sector, all approaches into San Francisco Bay, the East and San Pablo Bay area, and the Sacramento and San Joaquin River Delta, which encompasses the Sacramento and Stockton areas. VTS San Francisco consists of four remote radar sites, four communication sites, and five CCTV camera sites. Figure 2.4 shows VTS San Francisco's area of responsibility.

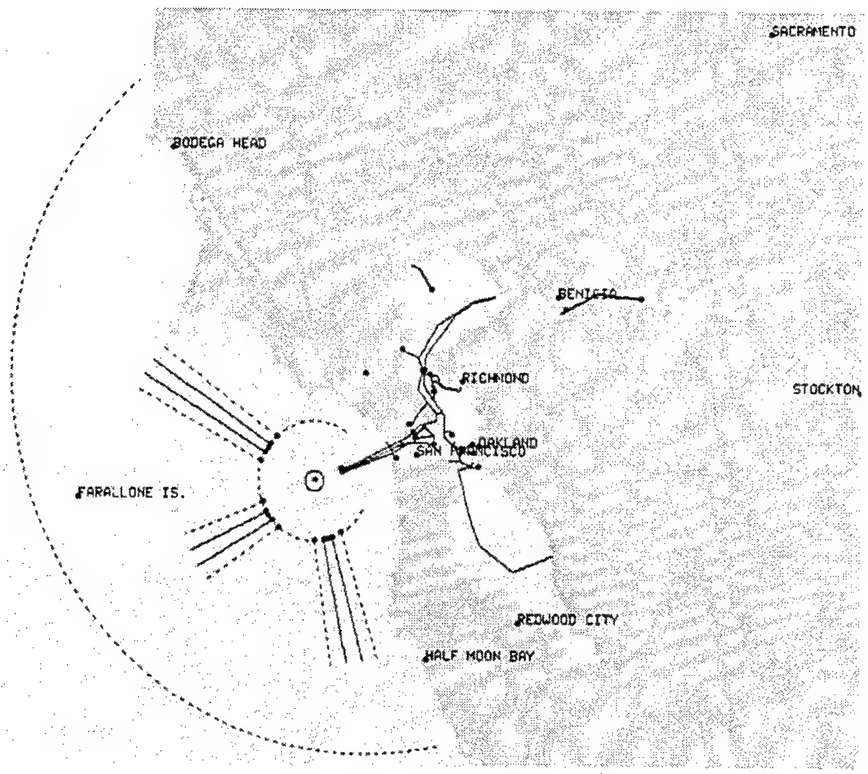


Figure 2.4. VTS San Francisco Area of Responsibility [7]

VTS San Francisco uses Raytheon 1342/SPS-64 radars for their Point Bonita and Yerba Buena Island sites and Furuno FR-8050D radars at their Mare Island and Point San Pablo radar sites. The Raytheon radar has a range resolution of 0.3% and a bearing resolution of ±0.3 degree while the Furuno radar is known to have a range resolution of 0.9% and a bearing resolution of ±1.0 degree. Table 2.2 shows the locations of the VTS San Francisco radar sites.

Radar Site	Site Location
Yerba Buena Island	37° 48' 34.71407 "N 122° 21' 55.31181" W
Point Bonita	37° 49' 12.08487" N 122° 31' 51.18769" W
Mare Island	38° 04' 10.23008" N 122° 15' 04.83339" W
Point San Pablo	37° 57' 43.55515" N 122° 25' 24.41049 W

Table 2.2. VTS San Francisco Radar Sites

B. TRACK DATA

At the operator display console, all vessels in a given remote site's AOR are displayed and depicted with the icon color designated for that site. For example, in the VTS San Francisco, target tracks colored "blue" represent reports from the Mare Island remote site, and "white" target tracks represent reports from the Point San Pablo remote site. In an area of overlapping radar coverage, both a white and a blue track are displayed for each known target. The tracks displayed to the operator are typically divided into three categories: radar tracks, automated dependent surveillance (ADS) tracks, and standard route (SR) tracks.

Radar tracks are the primary source of vessel data to the operator. Radar tracks are independently fused at the sensor level by the radar processor using a sequence of pairing, developing, and maturing operations [5]. Once a vessel is declared mature, the radar processor reports the vessel to the VTCS as an independent target. The report for a radar track continues until the target is either dropped by the operator or goes beyond the range of the sensor. Radar track reports are transmitted in run length encoded format from the RSS to the VTC and further processed to extract features or attributes (latitude, longitude, size, and time) for the database processor and the operator display console.

Automated dependent surveillance (ADS) tracks provide GPS and DGPS tracking capabilities to the VTS system. Vessels participating with ADS transmit their GPS or DGPS coordinates to the VTC via satellite or digital selective calling, which are then tracked and recorded by the database processor. ADS reports are not updated as frequently as radar track reports; however, they require minimal processing since the data are already formatted with latitude, longitude, time, course, and speed attributes.

Standard route tracks are synthetic tracks generated by the VTS system. SR tracks represent an estimated position (EP) of the vessel. SR tracks are generated once a target track is lost on a particular vessel. For instance, once a vessel leaves a site's AOR or if a site's sensor malfunctions, a SR track is displayed for that vessel. SR tracks are normally multi-segmented predefined routes fixed to the direction of the waterway. Vessels with predefined SR tracks typically transit the area frequently such as ferries. The SR track is terminated once the original or new sensor detects and tracks the target vessel. These tracks

inherit the same attribute format from the last radar or ADS track and require minimal format processing.

Each type of track is preprocessed into a format suitable for viewing and storing the path history of a vessel. The path history is recorded and archived in ASCII format and provides the essential attributes for the proposed fusion algorithm.

C. DATA COLLECTION

The use of actual vessel traffic data in this study validates the fusion algorithm's ability to handle "real-life" overlapping radar coverage scenarios and makes the algorithm suitable for use in existing VTS systems. Data sets were obtained from VTS Puget Sound and VTS San Francisco, which provided a variety of overlapping radar coverage scenarios.

1. VTS Puget Sound Data

Vessel traffic data were collected at VTS Puget Sound by the Inter-National Research Institute (INRI) and the USCG while conducting ADS trials during September 11-12, 1996. Overlapping radar coverage scenarios were collected and used in previous fusion algorithm studies [3,4]. The scenarios were found useful for implementation of the algorithm proposed in this study. The most evident area of overlapping radar coverage was in Puget Sound and Elliot Bay, as shown in Figure 2.5. This area has a high density of ferry traffic that transits from the downtown Seattle area to Winslow and Bremerton through Elliot Bay and Puget Sound. The two radars covering this area are the West Point radar, located west of Fort Lawton, and the Pier 36 radar, located at the northeastern point of Harbor Island.

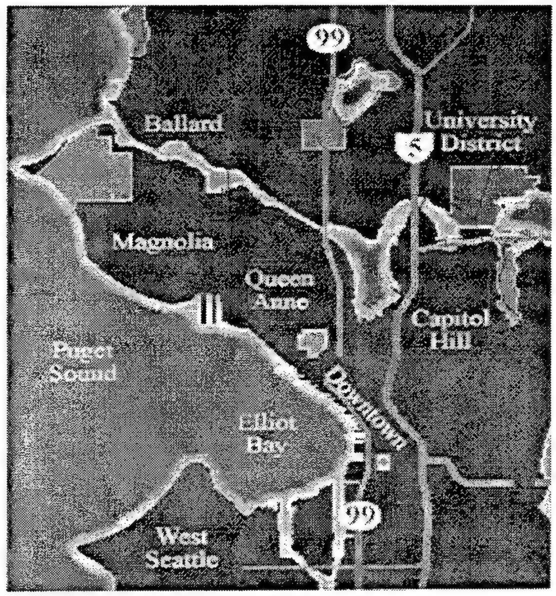


Figure 2.5. Puget Sound and Elliot Bay

The tracks of vessels transiting through this area were stored as ASCII files in comma delimited format. A sample of the contents of a recorded track file is listed below:

```
DAVID FOSS,120996230119,Radar,356,3,41.0,8.2,4735.20,-12225.23,252,0
UNK-4410,120996230119,Radar,409,3,91.3,17.6,4736.45,-12228.34,0,0
WALLA WALLA,120996230122,Radar,406,3,236.4,7.9,4736.08,-12223.72,439,0
SKAGIT,120996230122,Radar,405,3,48.5,13.9,4735.35,-12225.88,102,0
SPOKANE_ADS,120996230034,ADS,227,3669994520,92.0,17.9,4736.39,-12228.66,0,0
SPOKANE_ADS,120996230118,ADS,227,3669994520,89.4,18.4,4736.38,-12228.34,0,0
SPOKANE_ADS,120996230118,ADS,227,3669994520,89.4,18.4,4736.38,-12228.34,0,0
UNK-4409,120996230125,Radar,408,3,281.7,19.0,4736.78,-12225.02,0,0
DAVID FOSS,120996230125,Radar,356,3,42.4,8.2,4735.21,-12225.22,252,0
UNK-4411,120996230121,Radar,410,1,93.4,18.2,4736.45,-12228.16,0,0
UNK-4410,120996230125,Radar,409,3,92.1,17.7,4736.44,-12228.29,0,0
```

Each line in the above listing represents a reported track from a sensor sweep. The tracks are listed in time-sequence, and each line provides information about that track such as vessel name, position, sensor type, etc. Table 2.3 lists the specific features presented in each line.

1	2	3	4	5	6	7	8	9	10	11
cccc	DDMMYY hhmmss	AAA	cccc	cccc	x.x	x.x	ddmm.mm	ddmm.mm	size	x <CR>

Table 2.3. VTS Puget Sound Vessel Traffic Data Format

Column 1 contains the vessel's name in ASCII characters. If a vessel's name has not been determined, the vessel is referred to as unknown (UNK-XXXX). Column 2 is the Universal Time Code (UTC), which is the time of track position. Column 3 is the sensor type, which indicates whether the track is a radar, ADS, or SR track. Column 4 is the designated track ID number; each individual vessel is designated a unique ID number. Column 5 shows which sensor number is reporting the track. Column 6 shows the track's true course (in degrees), and column 7 indicates the speed (in knots over ground). Columns 8 and 9 are the track's latitude and longitude coordinates in degrees and minutes, respectively. Column 10 shows the calculated size of the vessel, and column 11 represents the quality of the track.

2. VTS San Francisco Data

Actual vessel traffic data were collected during the month of March 1999 at VTS San Francisco. The most evident area of overlapping radar coverage was the San Pablo Bay area, which consists of the Pinole Shoal Channel and the San Pablo Strait Channel as shown in Figure 2.6. The Mare Island and Point San Pablo radar sensors overlap this area, and vessel traffic data were collected from these sites.

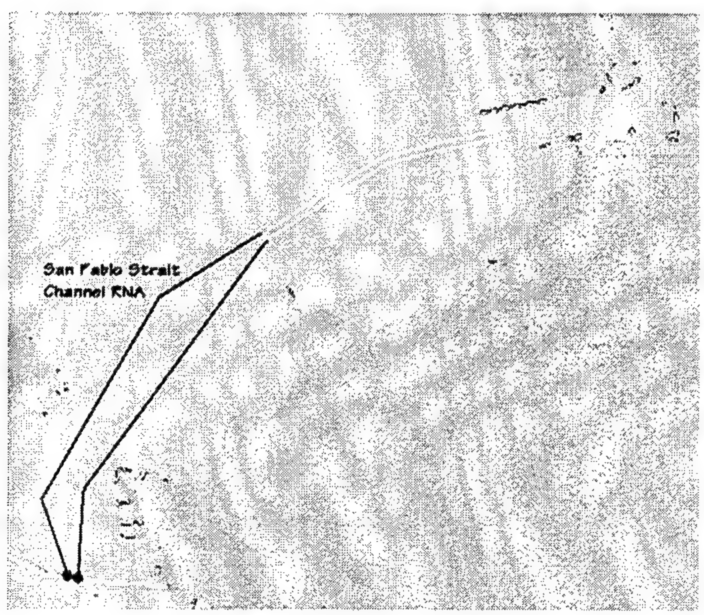


Figure 2.6. San Pablo Bay [7]

The method of data collection was different from that of VTS Puget Sound. Instead of using the data available from each sensor sweep at the display processor, vessel path histories from the archived database were used. Duplicate tracks are displayed at the operator's console, but only the sensor track designated by the operator is recorded as the vessel's path history and archived in the database. The other sensor track, the redundant track, is not archived. To obtain the data needed for the proposed algorithm, the redundant track was recorded as a "test" target to complement the path history of the actual vessel. VTS San Francisco personnel observed vessel traffic in this area and recorded track pairs for the purpose of this thesis.

Data were recorded in ASCII format with each line representing a track's position for each sensor sweep. A sample of the contents of a recorded duplicate track scenario is listed below:

MRI - ANNA FOSS	03/10/99-2253	SU	-122.36105	38.036331
MRI - ANNA FOSS	03/10/99-2256	SU	-122.37534	38.029651
MRI - ANNA FOSS	03/10/99-2259	RR	-122.38519	38.022498
MRI - ANNA FOSS	03/10/99-2300	SU	-122.38805	38.020695
MRI - ANNA FOSS	03/10/99-2303	SU	-122.39846	38.013381
MRI - ANNA FOSS	03/10/99-2307	SU	-122.40616	38.010261
PSP - TEST VESSEL	03/10/99-2253	TM	-122.3457	38.033465
PSP - TEST VESSEL	03/10/99-2253	MR	-122.36041	38.035842
PSP - TEST VESSEL	03/10/99-2256	SU	-122.37355	38.028171
PSP - TEST VESSEL	03/10/99-2300	SU	-122.38605	38.020973
PSP - TEST VESSEL	03/10/99-2303	SU	-122.39651	38.013725
PSP - TEST VESSEL	03/10/99-2307	SU	-122.40582	38.005948

Each of the lines above represents a reported track from a sensor sweep. Sensor site is indicated in the first column (MRI = Mare Island, PSP = Point San Pablo) followed by the vessel's name. The second column indicates the date and time of track followed by the status of the track. Track status codes are defined in Appendix A. The last column reports the longitude and latitude positions of the vessel in degrees.

The manual swapping of one sensor to another by the VTS operator in order to maintain the path history of a vessel is a task the proposed algorithm intends to automate. The algorithm will take the observed tracks from each sensor and the sensor resolutions of each site to determine the optimum sensor to track and record the path history of the vessel.

The next chapter discusses the concepts of multisensor/multitarget (MSMT) tracking systems and data fusion.

III. MULTISENSOR DATA FUSION

Multisensor data fusion techniques seek to combine data from multiple sensors to perform inferences that may be impossible to achieve based on data from a single sensor alone. Data may be in the form of positional coordinates (latitude and longitude), angular data (azimuth and elevation), or object identity declarations (such as Interrogation Friend or Foe (IFF)). The inferences so resolved from multisensor data fusion result in determination of a position or establishment of an identity.

Data fusion is beneficial for MSMT tracking systems. The combined data from multiple sensors ensures a more robust tracking system in the event a sensor fails. It also enables improved spatial coverage and detection capabilities compared to single sensor systems. For the VTS system, data fusion reduces redundant tracks displayed to the operator and determines the optimum sensor to track vessels. This chapter discusses the architectural model of the data fusion process used in tracking systems and the concepts of the positional fusion algorithm.

A. FUSION ARCHITECTURE

There are three basic types of data fusion architectures: centralized, autonomous, and hybrid. The type of architecture depends upon the complexity of the sensor and the processing needed to ensure the quality of estimates of the feature vectors [8]. Centralized fusion is the simplest of the three in terms of expense and computational efficiency. It requires only raw data from the sensors to be sent to the fusion center for processing, which results in minimal or no information loss. Autonomous and hybrid fusion architectures extract certain features from the raw data and provide sensor-level inferences of the data. The result is information loss due to the local optimization of the data from the sensors, prior to the multiple sensor fusion process.

VTS systems perform feature extraction at the sensor site in obtaining the range and bearing information of the target prior to transmission to the sensor data processors. The proposed fusion algorithm, presented in Chapter V, assumes that sensor level fusion is applied correctly at the remote sites; therefore, track information received at the VTC from

the sensor sites is treated as raw data, and central level fusion is applied. This simplifies the VTS fusion architecture into a centralized model as shown in Figure 3.1.

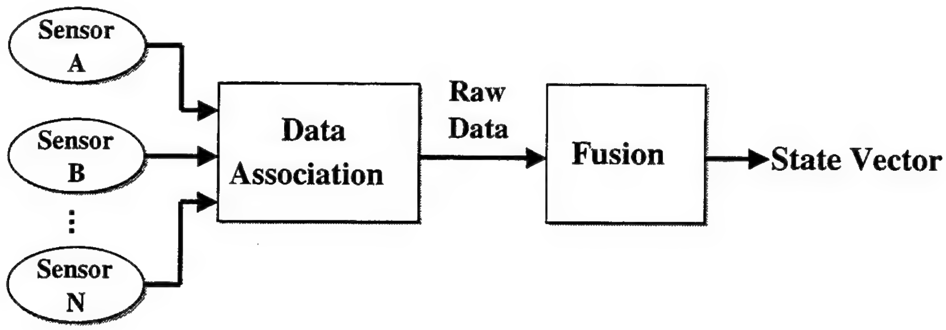


Figure 3.1. Centralized Fusion Model

In centralized fusion architectures, data from multiple sensors are used to estimate a target's position, velocity, and other attributes, or determine its identity [9]. The raw data from each sensor are associated using a data association process and then fused using physical models, pattern recognition techniques, or estimation techniques. The result is a state vector that establishes an identity declaration, such as an estimate of a position, or another attribute. In the VTS system, the raw data are the duplicate target tracks sent from a remote site. The fusion algorithm is applied to the track data, and the resulting state vector represents a unique track.

B. POSITIONAL FUSION

This study focuses on a centralized fusion model to reduce the redundant tracks in areas of overlapping radar coverage. An appropriate technique for the VTS system is positional fusion, which takes positional data from multiple sensors and infers an estimated or optimum position.

Positional fusion uses a combination of assigned thresholds and assumptions about the statistics of the noise processes to map observed data into a state vector, i.e., an independent set of variables such as latitude and longitude. This type of fusion provides an estimate of the state vector that best fits the observed data. Data from the sensors are preprocessed within a fusion system to perform data alignment, which transforms incoming

data into reference coordinates. After data alignment, two functions are performed: parametric association and state vector estimation. Figure 3.2 shows the flow diagram of positional fusion.

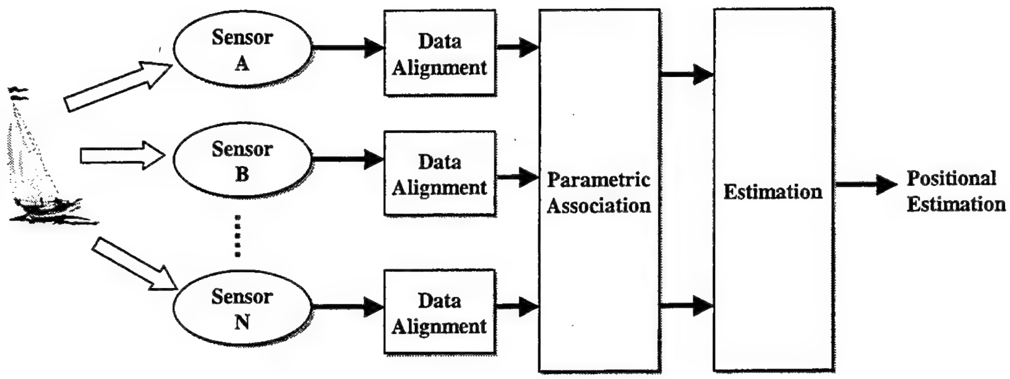


Figure 3.2. Positional Fusion

1. Sensor

Sensors for positional fusion algorithms provide spatial and temporal data for each target detected during a sensor scan. Sensors may be of different types but must convey information that can be understood by the fusion algorithm. A target's range and azimuth information from a radar sensor is preprocessed into positional coordinates and bearing from the sensor's known location. GPS and DGPS sensors also report the positional coordinates of a target. Sensor resolutions that must be taken into account during the association process. Chapter V describes how the sensor resolutions are utilized in the proposed algorithm.

2. Data Alignment

Data alignment orients the sensor data to a common spatial and temporal reference. This allows direct comparison of the data from each sensor for the association process. The spatial reference for the data in positional fusion algorithms depends upon the coordinate system in use. The spherical coordinate system of latitude and longitude in degrees is used in the VTS system. Temporal reference is achieved by aligning the sensor data to points in time. In real-time systems, data are referenced with respect to the current scan.

3. Data Association

Data or parametric association links observations from multiple sensors to individual targets by associating observations to other observations or existing tracks. This is accomplished by defining a measure of association that quantities the closeness between observation pairs. Association measures include correlation coefficients, distance measures, association coefficients, and probabilistic similarity measures. After an association measure is computed to determine the closeness between the two observations, association strategy or logic is applied to determine whether to declare the duplicate observations as one or more targets. Gating techniques establish boundaries or limits to provide an initial determination of whether the two observations could be physically related. In the VTS system, the resolution of the sensors would establish the bound or degree of uncertainty in which targets are duplicated or not. Figure 3.3 provides an example of positional association based on the uncertainty of the sensor.

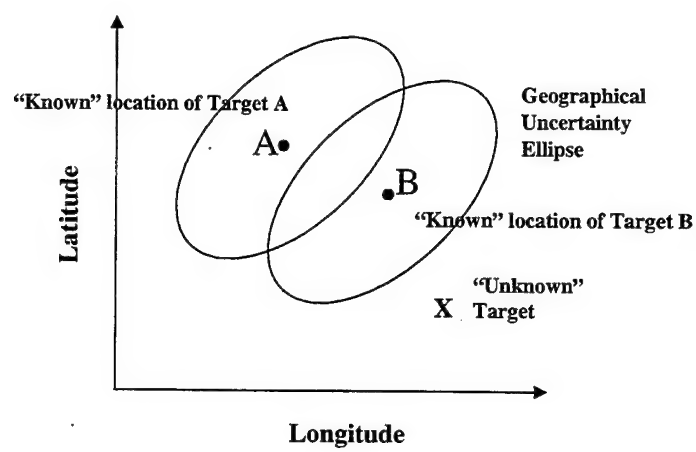


Figure 3.3. Positional Association and Sensor Uncertainties

In Figure 3.3, a new radar observation of "X" may fall within the uncertainty ellipse of Target A or B, both A and B, or neither A nor B. The geographical uncertainty ellipse around each target represents the area of possible error or range resolution of the sensor. From this observation, the following inferences can be made: X is associated with Target A; X is associated with Target B; X is neither associated with Targets A nor B; or X is a false target and should be ignored.

Given observation X and the known positions of Targets A and B, associations are formed that quantify the similarity between the observation and known position of the target. This similarity measure is then compared against a certain limit or *a priori* threshold. Associations are made using assignment logic, which may consist of hard or soft decision rules based on the similarity measures and thresholds. Chapter IV discusses the association process using assignment logic. After assigning the observation to the target, the fusion process then uses estimation techniques to fuse or combine the data to estimate the target's position.

4. Estimation

After the observations have been sorted by the association function, estimation techniques fuse the data. Estimation techniques determine the value of a state vector that best fits the observed data. Examples of estimation techniques are the least squares, weighted least squares, maximum likelihood, and minimum variance methods [10]. Estimation may also be performed using a track selection approach in which the most accurate sensor is chosen to track a vessel in a multisensor environment. An advantage of this form of estimation is that no composite or fused estimate is computed, cutting down on processing time and costs. The proposed algorithm uses this type of approach to establish the optimum target track among redundant reports. Chapter V discusses how the optimum sensor is established in the fusion algorithm.

In summary, multisensor data fusion is used to minimize the number of redundant target tracks displayed to VTS operators. In a centralized architecture, raw sensor data is sent to the fusion center for association and estimation. By aligning the input data from the sensors and associating them using similarity measures and threshold comparisons, fusion is performed to provide an estimate of a vessel's position as well as to determine the most reliable sensor for tracking.

IV. FUZZY ASSOCIATION

The proposed algorithm uses a fuzzy association technique to determine the optimum sensor to track targets in areas of overlapping radar coverage. The fuzzy association approach is preferred over traditional Boolean logic processes because it considers partial relationships or degrees of truth to the observed data, rather than the stringent relationship of "true" or "false". In a multisensor environment, a fuzzy association approach is suitable for determining the relationship between multiple track pairs. Instead of resolving track pairs as simply being correlated or non-correlated, they can be categorized into degrees of membership by quantifying the level of their correlation. From these degrees of correlation, a more informed assessment can be made whether or not to fuse the track pairs.

Figure 4.1 shows the flow diagram of the fuzzy association process. Data are sent from to the fusion center, where the degree of membership is assigned to the similarity measures of the data. The membership values are then applied to fuzzy association rules in which a decision is made about the data. This decision is then defuzzified to indicate fusion for the data received. This chapter discusses the advantages of using fuzzy logic for the association process, the design and application of membership functions, and the fuzzy association rules to establish fusion for the received data.

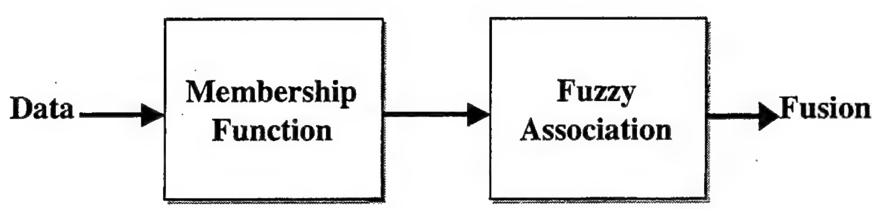


Figure 4.1. Fuzzy Association Process

A. FUZZY LOGIC

In classical set theory, an element may "belong" or "not belong" to a particular set. This can be described by the characteristic function χ_A of a set A in the universe of discourse U in which [11]

$$\chi_A(x) = \begin{cases} 1, x \in A, \\ 0, x \notin A. \end{cases} \tag{1}$$

This characteristic function has only two possible values: the statement *x belongs to A* is either "true" (a value of one) or "false" (a value of zero), for each element in U. The set A is referred to as a *crisp* set.

In fuzzy set theory or fuzzy logic, the set A can be extended into a *fuzzy* set. An element may still "belong" or "not belong" to a particular set, but may fall into categories in which an element belongs to the set as measured by a membership value. Descriptive or "fuzzy" categories such as *a lot*, *almost*, *like*, or *somewhat* represent how much or to what degree the element is part of the set. The relationship that an element makes with data can be modeled as the membership function μ_A, which represents the degree of truth in the statement *x belongs to A*:

$$0 \leq \mu_A(x) \leq 1, \textit{ for any } x \in U. \tag{2}$$

The difference between crisp and fuzzy logic theories is that the latter offers more information that can be used to derive a decision based on the data given to a fusion algorithm. In a multisensor tracking system, target tracks reported from different sensors have varying measures of similarity to each other. Using the data association example in Chapter III, Section B.3, the distance from Target A to Observation X provides a membership function for associating Observation X to Target A. The membership value assigned to the distance measure is a weight added to the decision-making in the association process. A membership value close to 1 indicates a strong correlation between X and A, and association between the two is heavily considered. A value at or near zero indicates a weak similarity, and association can be ignored.

Fuzzy logic offers several features that make it an excellent choice for data association techniques [12]. It is robust and does not require noise-free inputs from the sensors. The logic rules that govern the association are user-defined and can be refined for optimum performance. The system using fuzzy logic is also sensor dependent; however the sensors can be inexpensive and low in complexity. The leading advantage in using fuzzy logic over crisp logic is that fuzzy logic requires a smaller number of rule-based operations.

B. MEMBERSHIP FUNCTIONS

Membership functions are critical elements in the association process as they determine the degree of similarity and hence how the associations between elements and sets are made. As stated in Equation (2), membership functions take values in the range of 0 to 1, inclusive. A value of 1 represents a strong degree of truth, and a value of 0 represents a weak degree of truth. Membership values between zero and one are assigned either subjectively or from past evaluations and experiences. To illustrate the subjectivity in designing a membership function, the characteristic function for a crisp set shown in Figure 4.2 is compared to a membership function of a fuzzy set in Figure 4.3.

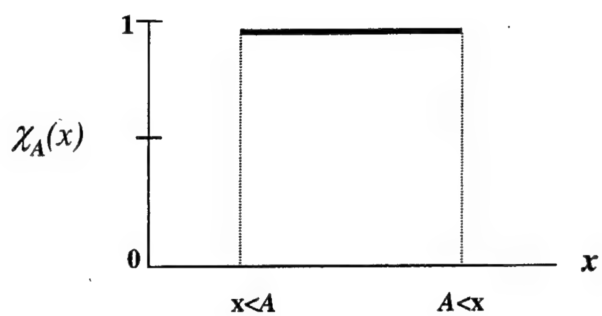


Figure 4.2. Characteristic Function for a Crisp Set

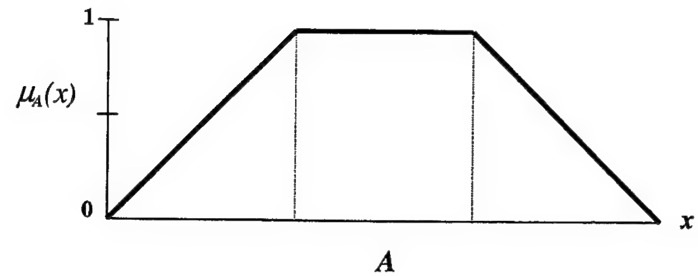


Figure 4.3. Membership Function for a Fuzzy Set

Membership functions can be depicted as graphical representations of the degree of truth of the elements to a set. As seen in the crisp set function, only the membership value of one can be applied to the element X if it is within the set A. In Figure 4.3, the membership value may range from zero to one depending on where the element X resides in A. The shape of the membership function is also subjective and different for each type of element. Figure 4.4 shows two membership functions used in previous VTS fusion algorithms for positional and course differences between target track pairs in a region of overlapping radar coverage [3,4].

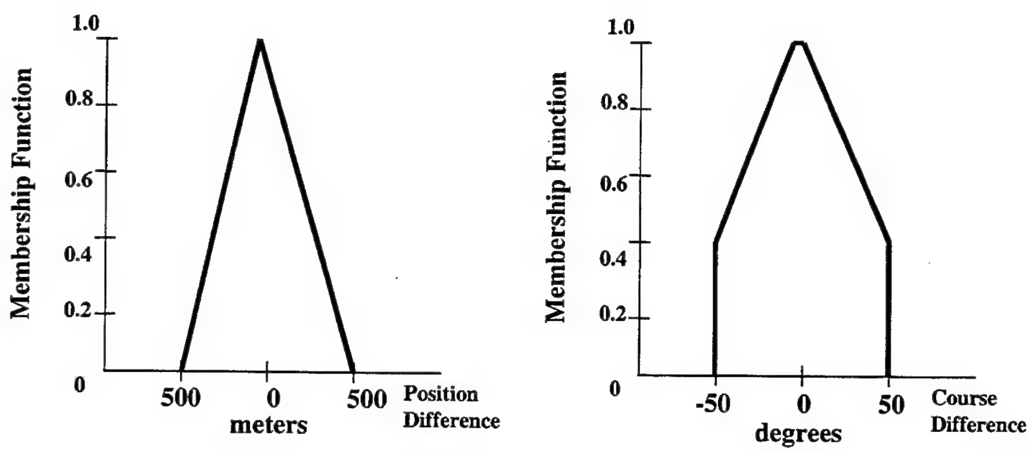


Figure 4.4. Membership Functions for Position and Course

To illustrate how membership functions are applied to positional differences between track pairs, the difference in latitude from one track to the latitude from another track is measured. The difference is measured in meters, and the membership value is

computed. If the difference is within 500 meters, a membership value from zero to one is assigned. The smaller the difference, the greater the degree of membership. For differences beyond 500 meters, it is inferred that the track pairs are separate targets. Once membership values are obtained for the data set, the association process begins.

C. FUZZY ASSOCIATION

Fuzzy association is ideal for multisensor tracking systems in that several associations can be made from the data received from each sensor. Fuzzy association uses IF-THEN-ELSE rules to establish association based on the membership values assigned to the similarity measure of the data and predetermined thresholds, α. IF-THEN rules are established so that if a membership value is greater than the threshold, the result is association, or else association is avoided:

$$\text{IF } \mu_A(x) > \alpha \text{, THEN Association}$$
$$\text{ELSE No Association}$$

After associating the input membership values, the membership function is defuzzified to obtain a crisp output. In control systems, this output provides feedback to the system. In multisensor tracking systems, the output is a decision whether or not the track pairs represent the same vessel. If the output indicates that the tracks belong to the same vessel, the fusion process then estimates the optimum sensor track. If the output indicates that the tracks belong to separate vessels, then the fusion algorithm repeats itself for the next set of data.

The fuzzy associative process for fusing redundant tracks is shown in Figure 4.5. Attribute data in the form of latitude, longitude, and course from one sensor are compared against the attribute data from another sensor to form a similarity measure. Once all of the assessed attributes for the track pairs have been assigned membership values, they are then checked against a designated threshold for that attribute. The threshold is selected by the operator or determined by the known resolutions of the sensor. If a threshold is not exceeded, the association for the track pairs fails, and further checks are stopped. If all

values exceed the assigned threshold, an association is made, indicated by a binary output of "1" from the defuzzifier, and the two target tracks are then fused to become one unique track displayed at the operator's console.

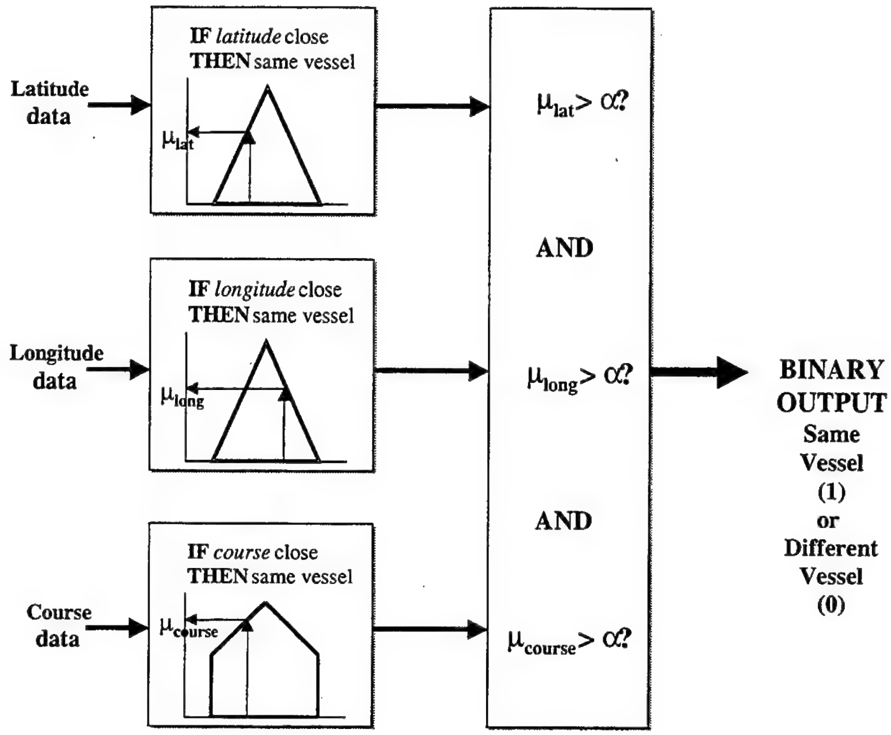


Figure 4.5. Fuzzy Associative System

Previous data association algorithms were successful in applying this fuzzy association technique to simulated and "real-life" target track data [3,4]; however, the membership function design was static and needed to be validated or optimized. An adaptive membership function that is shaped by the current statistics of the data would lead to a more accurate decision process. One such method that uses adaptive membership functions and relies on the current data is the *fuzzy clustering means algorithm.* The next chapter discusses the fuzzy clustering means algorithm and its application to the VTS environment.

V. FUZZY CLUSTERING MEANS ALGORITHM

The fuzzy clustering means (FCM) algorithm described in this chapter is used to first fuse the tracks based on the association of measurements and then select the sensor with the best measurement accuracy [1,2]. Only the tracks associated with the "best" sensor are displayed, resulting in a reduced number of tracks displayed to the VTS operator.

The FCM algorithm classifies data into groups or clusters by producing a degree of membership for each data point in the clusters. Data association is determined by selecting the highest degree of membership for each data-cluster pair. Because association is made among the membership values of the data in clusters and not with pre-selected threshold values, the FCM algorithm has fewer computations and is therefore simpler and less complex than traditional fuzzy association based algorithms. This chapter discusses the background for using the proposed fusion algorithm, the fuzzy clustering means algorithm for measurement-to-cluster assignment, and the FCM algorithm applied to the VTS environment for sensor-to-track association.

A. BACKGROUND

As discussed in Chapter IV, the design of membership functions for a fuzzy associative system is subjective and requires categorizing data elements into descriptive or *linguistic* variables, such as *very low*, *low*, *medium*, *high*, and *very high*. Increasing the number of linguistic variables for data elements increases the number of categories for the data. This results in an increase in precision. As the level of precision increases, the number of IF-THEN rules increases for the association process. The computational cost for optimal solutions becomes expensive as the number of measurements and variables increase. The required number of IF-THEN rules is given by

$$N_R = (lc)^s \tag{3}$$

where l is the number of linguistic variables, c is the number of measurements or sensors,

and s is the number of input variables. For example, to solve a tracking problem of associating six measurements with six tracks using only three input variables (latitude, longitude, and course) and five linguistic variables, the required number of IF-THEN rules is 27,000. Multisensor tracking systems using traditional fuzzy association are thus computationally expensive for tracking multiple targets with multiple sensors.

To reduce cost as well as the complexity for data association in MSMT environments, suboptimal solutions are used [1,2]. Suboptimal solutions do not require the precision of individual IF-THEN rules for every descriptive variable and measurement. To minimize the complexity of associating multiple tracks to multiple targets, measurements are processed into clusters and given a degree of membership within the cluster. This approach is called *fuzzy clustering*. The proposed algorithm uses fuzzy clustering to provide a suboptimal but computationally less expensive solution.

B. FUZZY CLUSTERING MEANS ALGORITHM

The FCM algorithm performs measurement-to-cluster association. Measurements are classified into clusters and compared to the cluster centers. The FCM algorithm consists of three main parts: calculating the similarity measures of the data, determining the fuzzy membership functions from the similarity measures, and applying the fuzzy rule system to the membership values for association. Figure 5.1 shows the diagram of the FCM algorithm.

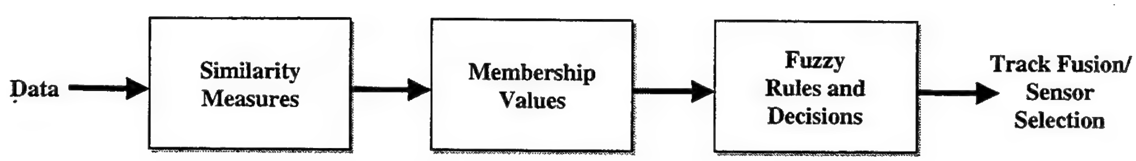


Figure 5.1. FCM Algorithm Diagram

1. Similarity Measures

The similarity measure d_{ik} is calculated from measurement x_k and the cluster center v_i for data point k and cluster i; it is the inner product induced norm between the two values and is represented as

$$d_{ik} = \|x_k - v_i\|. \tag{4}$$

Cluster centers are either predetermined or calculated from given membership values of the data. For positional fusion algorithms, cluster centers are the known geographical location of targets and are used as reference points for current or future measurements. Since the membership values are calculated from the measurements, the cluster centers are assumed fixed in this algorithm.

2. Membership Values

In fuzzy clustering, data point x_k is allowed to have a partial membership in more than one cluster. Let the partial membership value μ_{ik} represent the degree of membership of data point x_k in fuzzy cluster i. Given the number of clusters c and the number of data points n, the partial membership function is expressed as

$$\mu_{ik} \in [0,1], \quad 1 \le i \le c, 1 \le k \le n, \tag{5}$$

where the sum of all partial memberships for data point k in every cluster i equals 1

$$\sum_{i=1}^{c} \mu_{ik} = 1 \quad \forall k, \tag{6}$$

and the sum of all partial memberships for all data points in cluster i is between 0 and n

$$0 < \sum_{k=1}^{n} \mu_{ik} < n \quad \forall i. \tag{7}$$

Membership values are calculated using the similarity measures with respect to a fuzzification constant m. This constant reduces the influence of noise when computing the degree of membership or cluster centers. Given a cluster center v_i, the membership value for each data point k in cluster group i with respect to all cluster groups is defined as

$$\mu_{ik} = \frac{1}{\left[\sum_{j=1}^{c} (d_{ik} / d_{jk})^{2/(m-1)}\right]} \quad \forall i,k. \tag{8}$$

Once all partial membership values have been calculated, they are arranged in a partition matrix

$$U = \begin{bmatrix} \mu_{11} & \mu_{12} & \cdots & \mu_{1n} \\ \mu_{21} & \mu_{22} & \cdots & \mu_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \mu_{c1} & \mu_{i2} & \cdots & \mu_{cn} \end{bmatrix}, \tag{9}$$

where the columns of U represent the data points, and rows represent measured data points.

3. Association

The FCM algorithm performs measurement-to-cluster association by selecting the maximum membership value for each measurement-to-cluster pair. The approach consists of the following steps for n measurements received at time index or scan t:

1. Apply the FCM algorithm to find the partition matrix U. This matrix contains the membership values among all measurements and all targets.
2. Find the measurement-to-cluster pair with maximum membership value and assign measurement k to track i.
3. Remove the measurement-to-cluster pair identified in Step 2 (column k and row i) and obtain a reduced matrix.
4. Repeat Steps 2 and 3 for each of the remaining clusters until all n measurements are assigned to c existing clusters.

Once all measurements are assigned to a cluster, the process is repeated for the next sensor scan [1,2].

Consider an example of three targets ($c = 3$) all simultaneously scanned at time index t with fixed cluster centers v_1, v_2, and v_3 and measurements x_1, x_2, and x_3. Using Equations (4) and (8), the elements of the partition matrix U can be determined. For illustration, let us consider the following partition matrix:

$$U = \begin{bmatrix} \mu_{11} & \mu_{12} & \mu_{13} \\ \mu_{21} & \mu_{22} & \mu_{23} \\ \mu_{31} & \mu_{32} & \mu_{33} \end{bmatrix} = \begin{bmatrix} 0.32 & 0.65 & 0.22 \\ 0.44 & 0.17 & 0.23 \\ 0.24 & 0.18 & 0.55 \end{bmatrix}. \tag{10}$$

The max $[\mu_{ik}]$ is $\mu_{12} = 0.65$, and measurement 2 is assigned to cluster 1. The matrix is then reduced to

$$U_{reduced} = \begin{bmatrix} \mu_{21} & \mu_{23} \\ \mu_{31} & \mu_{33} \end{bmatrix} = \begin{bmatrix} 0.44 & 0.23 \\ 0.24 & 0.55 \end{bmatrix}, \tag{11}$$

where the max $[\mu_{ik}]$ of $U_{reduced}$ is $\mu_{33} = 0.55$, and measurement 3 is assigned to cluster 3. The matrix is then reduced to the last membership value where

$$U_{reduced2} = [\mu_{21}] = [0.44], \tag{12}$$

and the final assignment is measurement 1 to cluster 2.

C. FCM ALGORITHM IN THE VTS ENVIRONMENT

The FCM algorithm is modified for application to the VTS environment. Instead of measurement-to-cluster association, the VTS algorithm performs measurement-to-measurement association as well as the sensor selection for reducing duplicate tracks. The cluster centers v_i are unknown in this case, and association is performed using only the received data and sensor errors at scan t. This section discusses how the FCM algorithm is applied to the VTS environment to fuse the measurements and select the "best" sensor.

1. Similarity Measures

The total number of measurements n from sensor i are compared to measurements from sensor k. Measurement $\boldsymbol{x}_i$ is a report vector with p attributes from sensor i:

$$\boldsymbol{x}_i = \begin{pmatrix} attribute & 1 \\ attribute & 2 \\ \vdots & \\ attribute & p \end{pmatrix}, \quad i = 1,\ 2, \ldots, n. \tag{13}$$

Attributes from the sensors can be the latitude and longitude of the vessel, the course and speed calculated at the sensor level, or any combination of data that the VTS operator is required to measure. Chapter II lists the type of attributes that can be obtained from the sensor: latitude, longitude, true course, speed, UTC, track ID number, etc.

Since the similarity measure between $\boldsymbol{x}_i$ and $\boldsymbol{x}_i$ is zero, sensor errors are used to determine d_{ii}. Sensor errors provide a threshold to compare the similarity measures between reports. This threshold is as an uncertainty ellipse due to errors contributed by the sensor under consideration. The vector $\boldsymbol{e}_i$ represents the corresponding sensor errors of the attributes in $\boldsymbol{x}_i$:

$$\boldsymbol{e}_i = \begin{pmatrix} error & 1 \\ error & 2 \\ \vdots & \\ error & p \end{pmatrix}, \quad i = 1, 2, \ldots, n. \tag{14}$$

Examples of sensor errors are the position and course errors calculated from the radar's known range and bearing resolutions. For example, position error is due to the inaccuracies in the estimation of target location by the radar processor. The bearing resolution of a radar is used as the threshold for comparing the course differences between reports. Having the known sensor errors and the attribute differences between reports, the similarity measure d_{ik} between reports i and k and for the sensor attribute errors determined as follows:

$$d_{ik} = \begin{cases} \|\boldsymbol{x}_k - \boldsymbol{x}_i\|^2, if\ i \neq k \\ \|e_i\|^2, \quad if\ i = k \end{cases}, \quad i,k = 1,2,\ldots,n. \tag{15}$$

An $n \times n$ matrix containing the similarity measures of the sensor errors (along the diagonal elements) and the similarity measures of the measurements is formed:

$$D = \begin{bmatrix} \|e_1\|^2 & \|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2 & \cdots & \|\boldsymbol{x}_1 - \boldsymbol{x}_n\|^2 \\ \|\boldsymbol{x}_2 - \boldsymbol{x}_1\|^2 & \|e_2\|^2 & \cdots & \|\boldsymbol{x}_2 - \boldsymbol{x}_n\|^2 \\ \vdots & \vdots & \ddots & \vdots \\ \|\boldsymbol{x}_n - \boldsymbol{x}_1\|^2 & \|\boldsymbol{x}_n - \boldsymbol{x}_2\|^2 & \cdots & \|e_n\|^2 \end{bmatrix} = \begin{bmatrix} d_{11} & d_{12} & \cdots & d_{1n} \\ d_{21} & d_{22} & \cdots & d_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ d_{n1} & d_{n2} & \cdots & d_{nn} \end{bmatrix}. \tag{16}$$

2. Fuzzy Membership Calculations

Membership values μ_{ik} are calculated using Equation (8), and the partition matrix U as in Equation (9) is formed. The diagonal elements μ_{ii} represent the membership values of the sensor errors of sensor i, and the off-diagonal elements μ_{ik} represent the membership values of the similarity measures between reports $\boldsymbol{x}_i$ and $\boldsymbol{x}_k$.

3. Fuzzy Rules and Decision System

After applying the FCM algorithm to obtain membership values for all measurements and sensor errors, the highest membership value indicates the appropriate association of measurement i to measurement k. The association is made using a decision rule that compares the membership value μ_{ik} to that of the sensor error μ_{ii}. This association indicates whether the tracks are from the same target or separate entities.

The decision rule H_i is based on the elements partition matrix U for each sensor i:

$$H_i = \begin{cases} 1, if\ \mu_{ik} > \mu_{ii} \\ 0, if\ \mu_{ik} < \mu_{ii}. \end{cases} \tag{17}$$

A binary decision of 1 means that measurements i and k are within the boundaries of the sensor accuracy; the tracks are declared to belong to the same target and then fused. A decision of 0 means that the measurements exceed the sensor accuracy, and the tracks are not fused, thus treating them as separate targets.

Once track correlation has been established, defuzzification of the membership values μ_{ii} results in the selection of the most accurate sensor for tracking. The selected sensor has the maximum degree of membership (max $[\mu_{ii}]$), and the track reported from that sensor is considered the fused track. The track of the selected sensor is then displayed to the VTS operator, and the FCM algorithm is repeated for the next set of sensor reports.

This algorithm is more efficient than traditional fusion algorithms because the algorithm relies on present data and known sensor resolutions. This saves on memory and on processing time since past information does not have to be retrieved. In addition, positional estimation is not performed since the algorithm selects the superior sensor for tracking.

In summary, the FCM algorithm for the VTS environment uses decision rules based upon the magnitudes of the membership values to fuse measurements and select the most accurate sensor for tracking. This adaptive strategy is more convenient than conventional fuzzy association techniques since membership values are compared against each other and not to fixed membership functions for each attribute. The result is a more efficient and less computationally complex algorithm to reduce redundant tracks displayed in a MSMT tracking system.

VI. SIMULATION

Actual vessel traffic data from VTS Puget Sound and VTS San Francisco were used to test the algorithm. Overlapping radar coverage scenarios with duplicate target tracks in both sets of data were extracted for implementation in MATLAB® software. This chapter discusses the preprocessing of the data, determination of the sensor's range error, and the results obtained from test scenarios.

A. DATA PREPROCESSING

Preprocessing the data from VTS Puget Sound to test the algorithm is accomplished using the *getdatax.m* function, previously created for formatting text data into a suitable matrix form for MATLAB® [3,4]. The code for this function is listed in Appendix B. The resulting matrix has ten columns; the number of rows is the number of observations. The columns are:

ObsnMatrix = [**Latitude**
Longitude
TrueCourse
Speed
Size
TrackIDNumber
UTC
TrackQuality
TrackStatus
SensorTrackNumber].

Preprocessing the data from VTS San Francisco required formatting the positional attributes into matrix form as well. Since tracks were recorded simultaneously in time, the position elements were extracted as a separate data file:

ObsnMatrix = [**Latitude**
Longitude].

B. SIMULATION CONSTRUCTION

The goal of this thesis is to fuse redundant target tracks in real-time. As sensor information is received at the VTC, the fusion process occurs, and the selected sensor track is displayed to the VTS operator. The fusion process is transparent to the operator and continues until the target is no longer in the area of overlapping radar coverage.

In order to achieve real-time fusion and data association, the vessel traffic data is temporally aligned. The collected data from each VTS contain a time field for each observation. VTS Puget Sound data contain the UTC in column 2 of each observed track row. VTS San Francisco data contain the time field in column 3 of each observed track. Sensor reports observed at time DDMMYYhhmmss or MM/DD/YY-hhmm are compared and then used in the FCM algorithm. Here is an example of sequenced and paired data for three observations at time 1109962119:

UNK-4773,**110996211940**,Radar,772,3,117.2,18.2,4736.42,-12229.03,0,0
SPOKANE_ADS,**110996211941**,ADS,773,3669994520,98.2,17.5,4736.38,-12229.02,0,0
UNK-4775,**110996211945**,Radar,774,1,105.6,17.8,4736.44,-12228.80,0,0

In all cases, tracks are aligned to the nearest minute. The attributes from each sensor report are compared with the attributes of other reports at that time index. With the data aligned in time, observations are processed as if the data are received in real-time from each sensor scan.

C. RANGE RESOLUTION

The range resolution given for each radar type was a percentage of the distance from the sensor to the tracked target. To factor the sensor's range resolution into the FCM algorithm, the distance from the sensor to the target needed to be derived. Since the positional coordinates of each target were reported in each data set, the range (measured in nautical miles (NM)) from the known position of the sensor was determined using a spherical-coordinate calculation. Let ϕ_1 and θ_1 represent the sensor's latitude and longitude coordinates in degrees, and ϕ_2 and θ_2 represent the target's latitude and longitude coordinates in degrees. Using 3443.9 NM as the approximate radius of the earth r_{earth} [13],

the range calculation is as follows [14]:

$$\text{Range} = r_{earth} \arccos(\cos\phi_1 \cos\phi_2 \cos(\theta_2 - \theta_1) + \sin\phi_1 \sin\phi_2). \quad (18)$$

Once the range is determined in nautical miles, the range error of the sensor in degrees is determined. Given a range resolution of 0.375% (Raytheon radar), at a measured range of 12 NM, the ranger error is determined to be 0.045 NM or 91.14 yards. Because the VTS data are available in coordinates of latitude and longitude, the range error is expressed in units of degrees and minutes. Each degree of latitude equals 60 NM; equivalently, one minute represents one nautical mile. We have

$$\text{Range Error in degrees Latitude} = \text{Range Error in NM} \div 60. \quad (19)$$

Because of the oblong curvature of the earth, each degree of longitude results in a distance equal to or less than 60 NM. At the Equator (00° Latitude), one degree Longitude represents 60 NM. As we move away from the Equator towards the North or South Pole, a degree in longitude represents less than 60 NM, proportional to the distance from the Equator. In the Puget Sound vicinity, one nautical mile equals approximately 1.5 minutes Longitude. In the San Francisco Bay area, one nautical mile is equal to approximately 1.25 minutes Longitude. The following expressions can be used to convert the range errors from nautical miles to degrees Longitude:

$$\text{Range Error in degrees Longitude} = \text{Range Error in NM} \div 40 \text{ (Puget Sound)}, \quad (20)$$

$$\text{Range Error in degrees Longitude} = \text{Range Error in NM} \div 48 \text{ (San Francisco)}. \quad (21)$$

Once the range errors are converted to degrees Latitude and Longitude, they are applied to the fusion algorithm as the sensor error vector $\boldsymbol{e}_i$ for the latitude and longitude attributes reported in $\boldsymbol{x}_i$ (see Chapter V).

D. NUMERICAL EXAMPLE

The following example illustrates the application of the fusion algorithm presented in Chapter V to data obtained from the VTS San Francisco Mare Island and Point San Pablo sites. This example considers one time scan of two sensors reporting two measurements of two tracks. Furuno radars are used at these two sites with a known range resolution of 0.9%. This is a systematic example of how the fusion algorithm is applied to VTS data in the MATLAB® code.

1. Similarity Measures and Sensor Accuracy

The algorithm begins with the two sensor sites transmitting the positional coordinates (latitude and longitude in degrees) of each track to the VTC. Let $\boldsymbol{x}_1$ represent reports from the Mare Island site and $\boldsymbol{x}_2$ from the Point San Pablo site:

$$x_1 = \begin{bmatrix} 38.0394 \\ -122.3408 \end{bmatrix} \qquad x_2 = \begin{bmatrix} 38.0398 \\ -122.3403 \end{bmatrix}.$$

Using Equation (15), similarity measures d_{12} and d_{21} are computed to be 0.0007°. The next step is to find the sensor errors d_{11} and d_{22}.

Because the range resolution of the sensor is given as a percentage, the ranges from the sensors to the target need to be determined. This is done for each sensor using Equation (20). For the Mare Island site: ϕ_1 = 38.0717° Latitude, θ_1 = -122.2508° Longitude (Table 2.2), ϕ_2 = 38.0394° Latitude, and θ_2 = -122.3408° Longitude. The calculated range from the target to the Mare Island site using Equation (18) is 4.6833 NM. Using the positional coordinates for the Point San Pablo site and the target, the calculated range is 6.3522 NM.

Using Equations (19) and (21), the Mare Island site's range error is determined to be 0.0007° Latitude and 0.0009° Longitude. The Point San Pablo site's range error is calculated to be 0.0010° Latitude and 0.0012° Longitude. The range errors (in degrees) can now be presented as error vectors $\boldsymbol{e}_1$ and $\boldsymbol{e}_2$ for the Mare Island sensor and Point San Pablo sensor, respectively:

$$e_1 = \begin{bmatrix} 0.0007 \\ 0.0009 \end{bmatrix} \text{and } e_2 = \begin{bmatrix} 0.0010 \\ 0.0012 \end{bmatrix}.$$

Similarity measures of sensors errors d_{11} and d_{22} are then calculated by using Equation (15). The similarity measures of the data and the sensor errors are then placed in the symmetric matrix D:

$$D = \begin{pmatrix} \|e_1\|^2 & \|x_2 - x_1\|^2 \\ \|x_1 - x_2\|^2 & \|e_2\|^2 \end{pmatrix} = \begin{pmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{pmatrix} = \begin{bmatrix} 0.0011 & 0.0007 \\ 0.0007 & 0.00152 \end{bmatrix}.$$

This matrix provides the elements for the fuzzy membership calculations using the FCM algorithm.

2. Fuzzy Membership Calculations

The membership values μ_{ik} and μ_{ii} for the similarity measures d_{ik} and sensor errors d_{ii} are calculated using Equation (8); a fuzzification constant of $m = 2$ is used. This results in a partition matrix with the membership values:

$$U = \begin{pmatrix} \mu_{11} & \mu_{12} \\ \mu_{21} & \mu_{22} \end{pmatrix} = \begin{bmatrix} 0.2589 & 0.8404 \\ 0.7411 & 0.1596 \end{bmatrix}.$$

3. Fuzzy Association

From the partition matrix U, the association is carried out by applying the decision rule H_i from Equation (17) to the membership values of the sensor errors and the similarity measures of the two tracks. For the Mare Island sensor, $H_1 = 1$ ($\mu_{12} > \mu_{11}$) and for the Point San Pablo sensor, $H_2 = 1$ ($\mu_{21} > \mu_{22}$).

The decision rule determines that the two tracks are the same, and the sensor with the highest membership value is selected as the sensor to track the target. In this case, μ_{11} is selected ($\mu_{11} > \mu_{22}$). The superior sensor is the Mare Island radar. The report from Point

San Pablo is discarded, and the track displayed to the VTS operator is the observed track from the Mare Island site for this scan. The operator is notified that correlation and fusion have occurred, and the track is then recorded as the vessel's path history and archived into the database. The algorithm resets itself for the next surveillance scan.

E. TESTS

Several test scenarios were applied to the fusion algorithm using MATLAB® Version 5.3 running on a Windows platform. All scenarios featured overlapping sensor coverage from multiple sensors, which produced multiple radar tracks. Radar tracks are plotted before and after fusion to demonstrate the effectiveness of the algorithm. Each test was performed independently using the MATLAB® code listed in Appendix C.

1. VTS Puget Sound Scenarios

Three types of scenarios of overlapping sensor coverage were tested. The first test involved overlapping coverage of two sensors reporting two tracks with two attributes (latitude and longitude). The second scenario tested the algorithm with two sensors reporting two tracks with the addition of a third attribute (course). The third test presented the algorithm with three sensors reporting three tracks with latitude and longitude as attributes. Each scenario was monitored by the West Point and Pier 36 radar sites. The AIL FPS-109 radar is used at both sites, with known range and bearing resolutions of 0.375% and ±0.35 degrees, respectively.

a. Two Sensors, Two Tracks, and Two Attributes

The first test involves overlapping radar coverage of tracks 750 and 751. From observing the data sequenced in time, the tracks are on a northbound course through Puget Sound, approaching the West Point site. Figure 6.1 shows the relative position of the tracks to the location of the sensors. Sensor 1 is the West Point site, and Sensor 2 is the Pier 36 site. Track 750 is from Sensor 1, and track 751 is reported by Sensor 2.

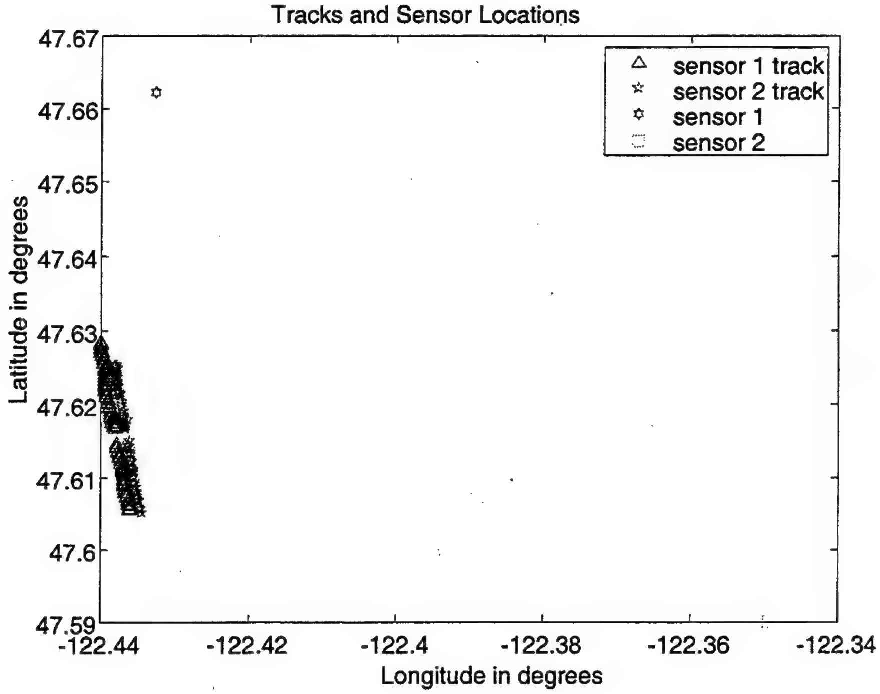


Figure 6.1. VTS Puget Sound Tracks 750 and 751 with Sensor Sites:
No Fusion Applied

The longitudinal separation between the two tracks was approximately 0.07 nautical miles or 142 yards. The algorithm determined that the two tracks belong to the same vessel and selected the West Point site as the sensor of choice throughout the duration of the data set. Figure 6.2.a shows the duplicate tracks before fusion, and Figure 6.2.b shows the track after fusion. The latter plot is displayed to the VTS operator.

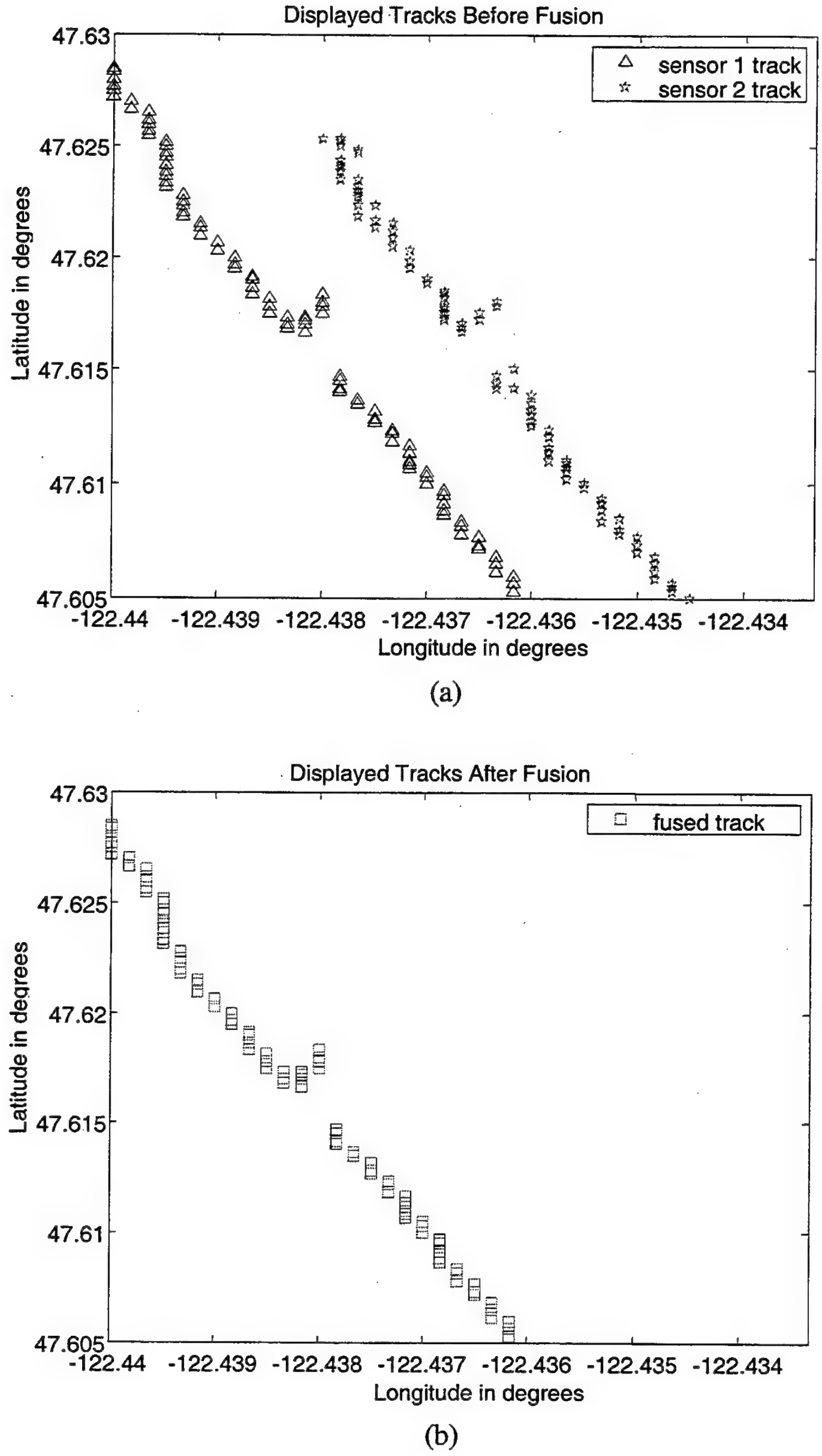


Figure 6.2. VTS Puget Sound Tracks 750 and 751, Two Sensors, Two Tracks, and Two Attributes (latitude and longitude): (a) no fusion applied and (b) fusion applied

b. Two Sensors, Two Tracks, and Three Attributes

The second test involved overlapping radar coverage of tracks 830 and 831. Three attributes (latitude, longitude, and bearing) from each sensor were considered. Bearing resolution was taken into account for the course attribute. Observing the data in time, the tracks indicate an eastbound course through Elliot Bay towards downtown Seattle. Figure 6.3 shows the observed data points for these two tracks in relative position to the two sensors. Sensor 1 is the West Point site, and Sensor 2 is the Pier 36 site; 830 is track 1, and 831 is track 2.

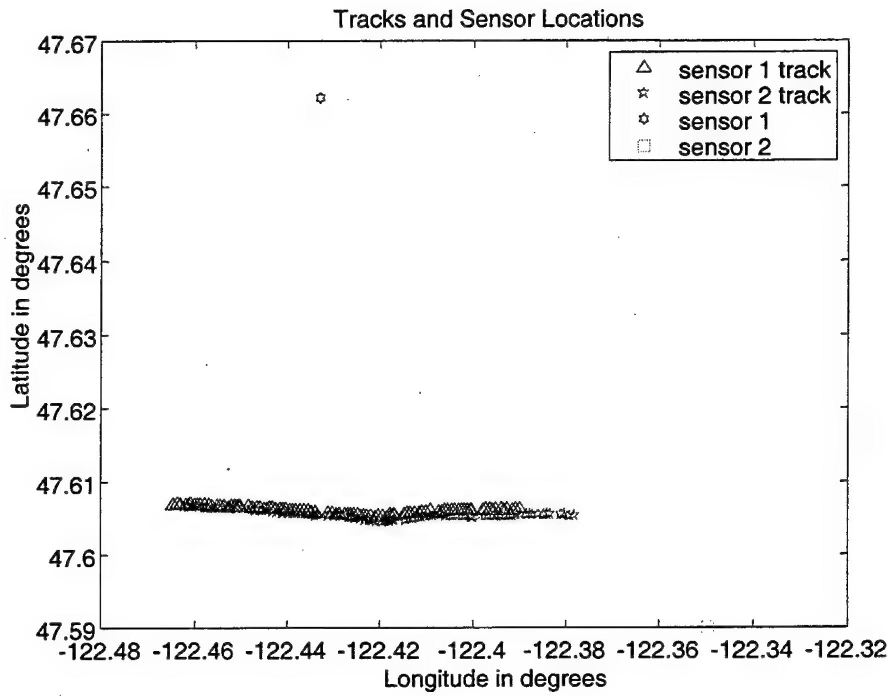
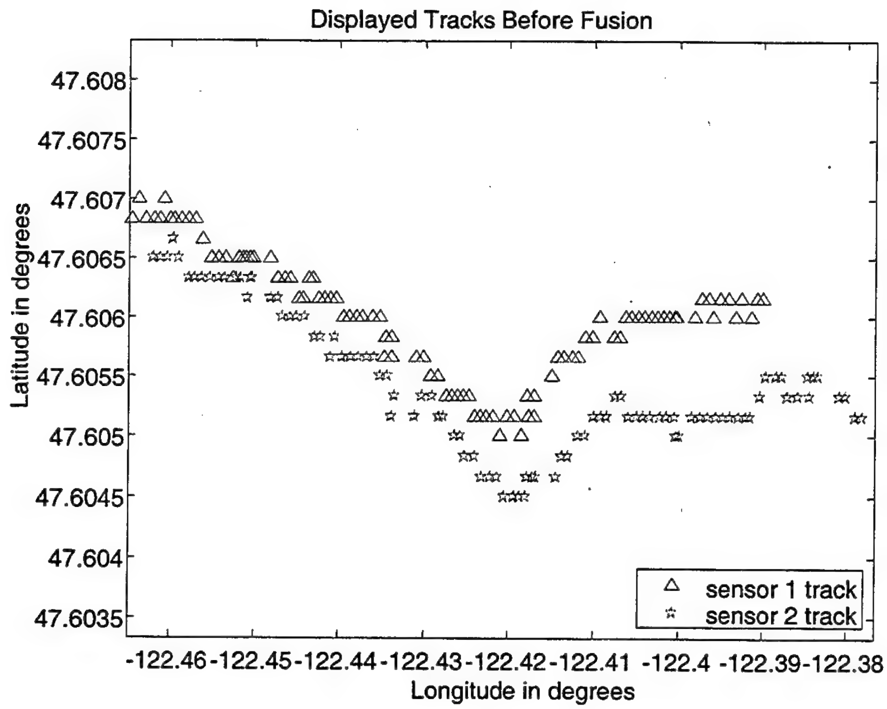
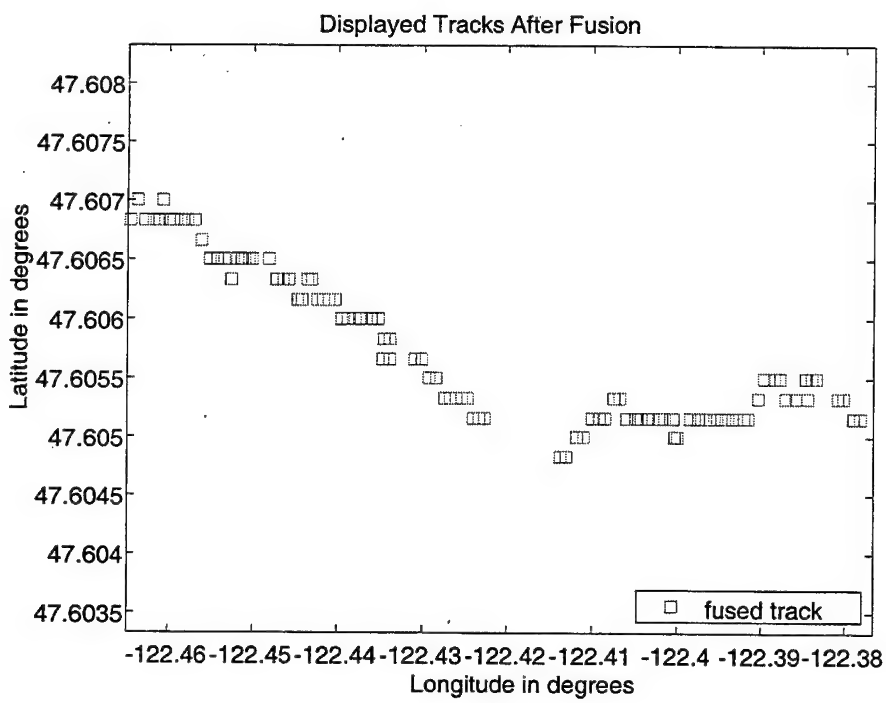


Figure 6.3. VTS Puget Sound Tracks 830 and 831 with Sensor Sites: No Fusion Applied

After applying the algorithm to tracks 830 and 831, the two tracks were fused to be from one vessel. The selected track was initially determined to be the Sensor 1 track (Track 830) but was handed off to the Sensor 2 track (Track 831) as the vessel approached Pier 36 radar. The hand off was approximately 3.47 NM from the West Point site and 2.73 NM from the Pier 36 site. Figure 6.4 shows the tracks before and after fusion.



(a)



(b)

Figure 6.4. VTS Puget Sound Tracks 830 and 831, Two Sensors, Two Tracks, Three Attributes (latitude, longitude, and bearing): (a) no fusion applied and (b) fusion applied

c. *Three Sensors, Three Tracks, and Two Attributes*

The third test involved overlapping sensor coverage of tracks 772, 773, and 774. Two tracks were measured using two separate radars. The third track is an ADS track in which GPS information is given to the VTC by the participating vessel. GPS sensor resolution is approximately ±22.965 yards. Using Equations (39) and (40), the resolution was converted from nautical miles to degrees Latitude and Longitude. The tracks indicate an eastbound course. Figure 6.5 shows the three tracks relative to the location of the West Point site (Sensor 1) and the Pier 36 site (Sensor 2).

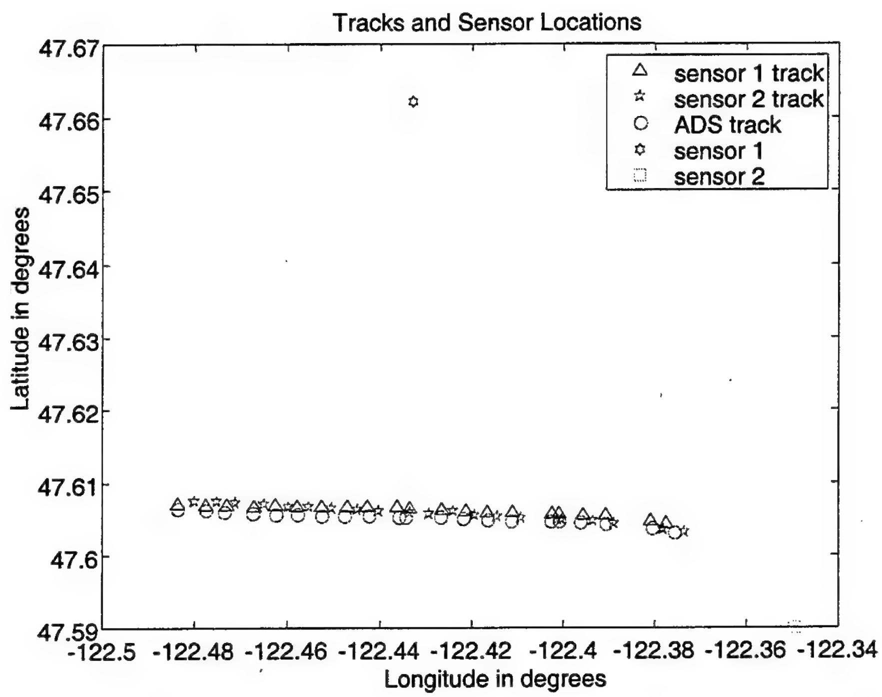


Figure 6.5. VTS Puget Sound Tracks 772, 773, and 774 with Sensor Sites: No Fusion Applied

When the fusion algorithm was applied to these tracks, the tracks were fused to be from the same vessel, and the selected track based on sensor resolution was the ADS track. At 3.37 NM from West Point and 3.80 NM from Pier 36, however, the ADS track was swapped for Sensor 2's track (Pier 36). After two sensor scans, the ADS track resumed the role of the selected track. An outlying track point from Sensor 1 was selected for one scan index; no specific reason could be attributed to this. Figure 6.6 shows the tracks before and after fusion.

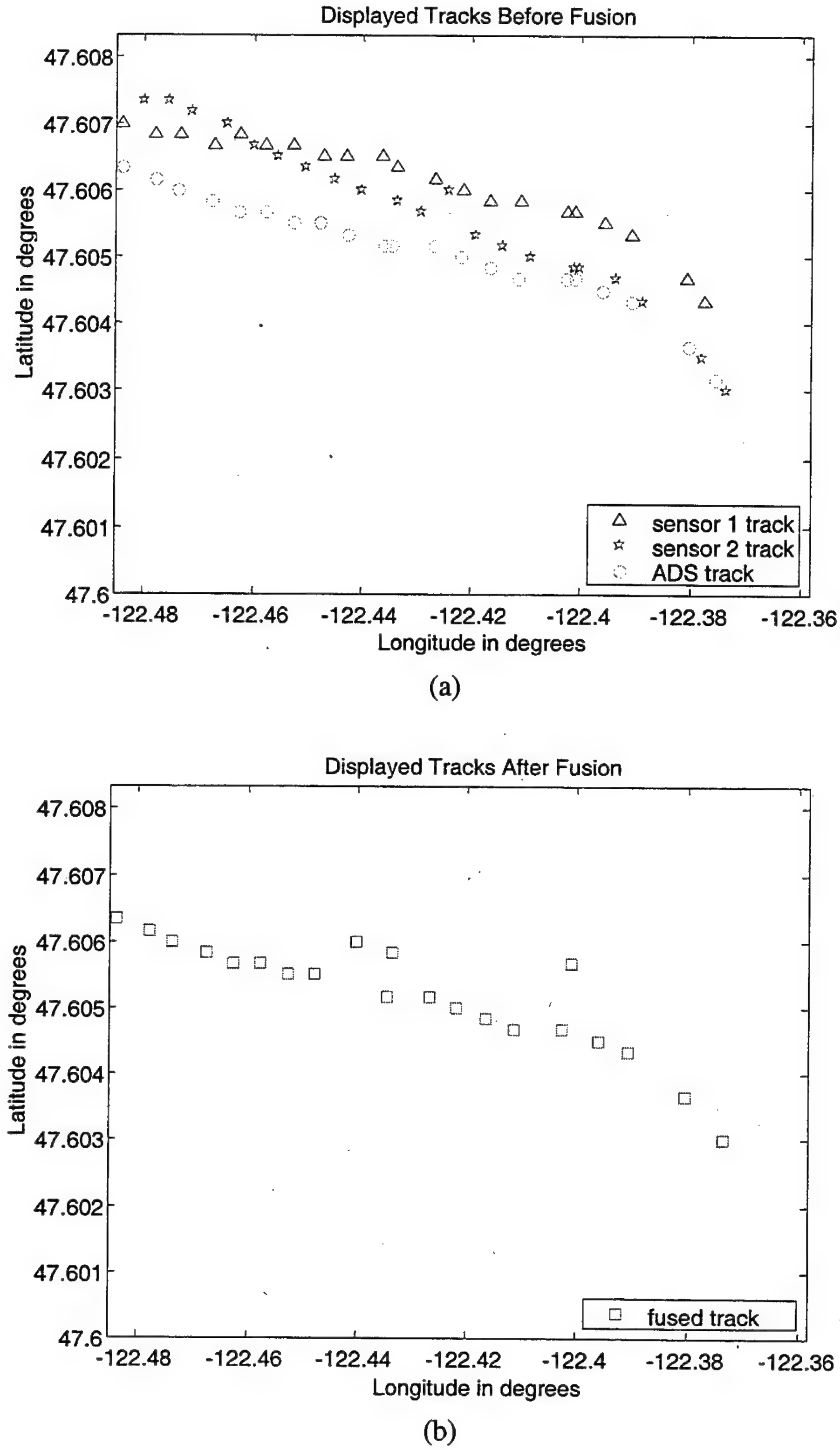


Figure 6.6. VTS Puget Sound Tracks 772, 773, and 774, Three Sensors, Three Tracks, Two Attributes (latitude and longitude): (a) no fusion applied and (b) fusion applied

2. VTS San Francisco Scenarios

The most evident area of overlapping radar coverage in VTS San Francisco is the San Pablo Bay area, monitored by the Mare Island and Point San Pablo radar sites. Several track scenarios were tested using the algorithm and consistent results of duplicate radar track fusion were obtained for all data sets. The scenarios were composed of two sensors reporting two tracks consisting of two attributes (latitude and longitude). Both sensor sites use the Furuno FR-8050D radar with a known range resolution of 0.9%.

The first scenario involved the vessel SAN JOAQUIN on a southbound course through the Pinole Shoal and San Pablo Strait channels. SAN JOAQUIN's path history was initially tracked by the Mare Island sensor site (Sensor 1). Recorded observations commenced once the Point San Pablo sensor detected the target and displayed a duplicate track. Figure 6.7 shows the observed tracks in relation to the location of the two sensors.

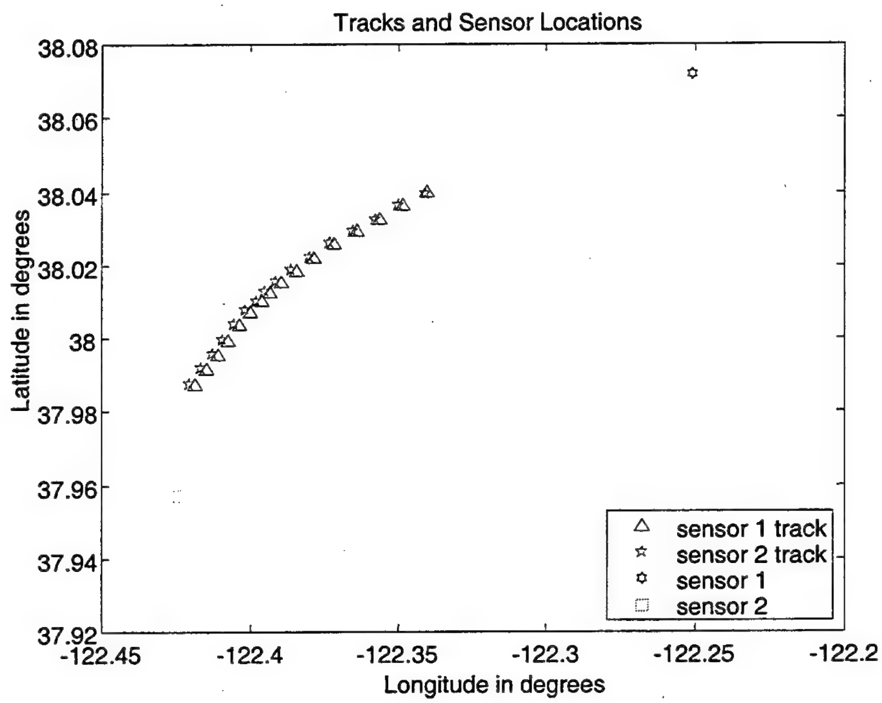


Figure 6.7. VTS San Francisco Track SAN JOAQUIN with Sensor Sites: No Fusion Applied

Figure 6.8.a shows the displayed tracks before the fusion algorithm was applied. The two tracks have an approximate longitudinal separation of 0.096 NM or 194 yards. After applying the fusion algorithm to the data set, the two tracks were fused to be from one vessel, and the Mare Island sensor was initally selected as the superior sensor. After 5.5 NM from the Mare Island site and 5.5 NM from the Point San Pablo site, the Mare Island sensor was swapped for the Point San Pablo sensor. The Point San Pablo radar was selected for the remaining scans as the tracks approached the sensor site. Figure 6.8.b shows the track displayed to the VTS operator after the fusion algorithm was applied.

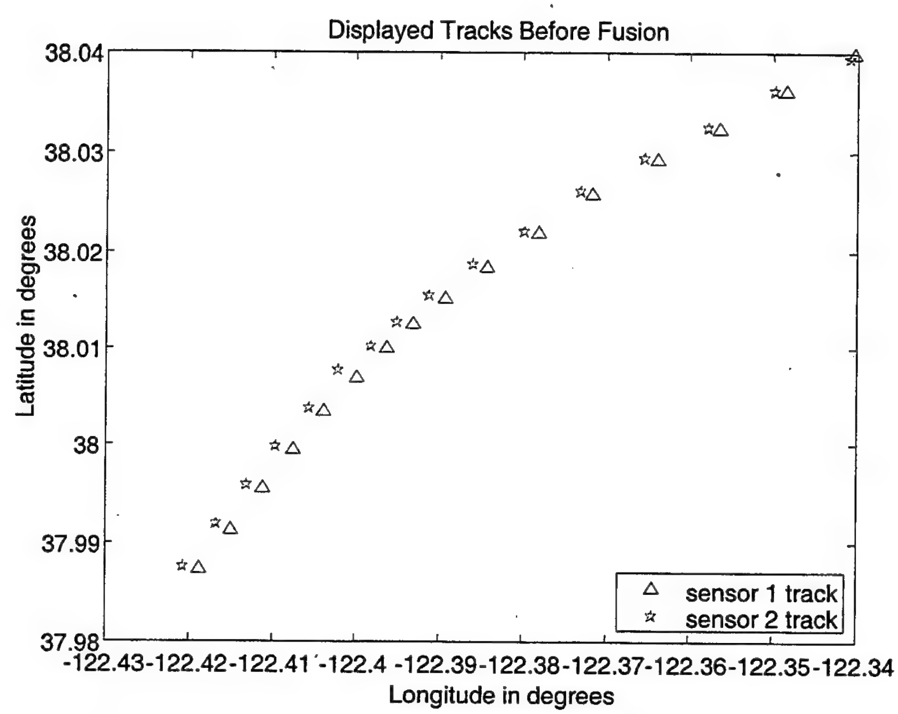


Figure 6.8.a. VTS San Francisco Track SAN JOAQUIN, Two Sensors, Two Tracks, Two Attributes (latitude and longitude): no fusion applied

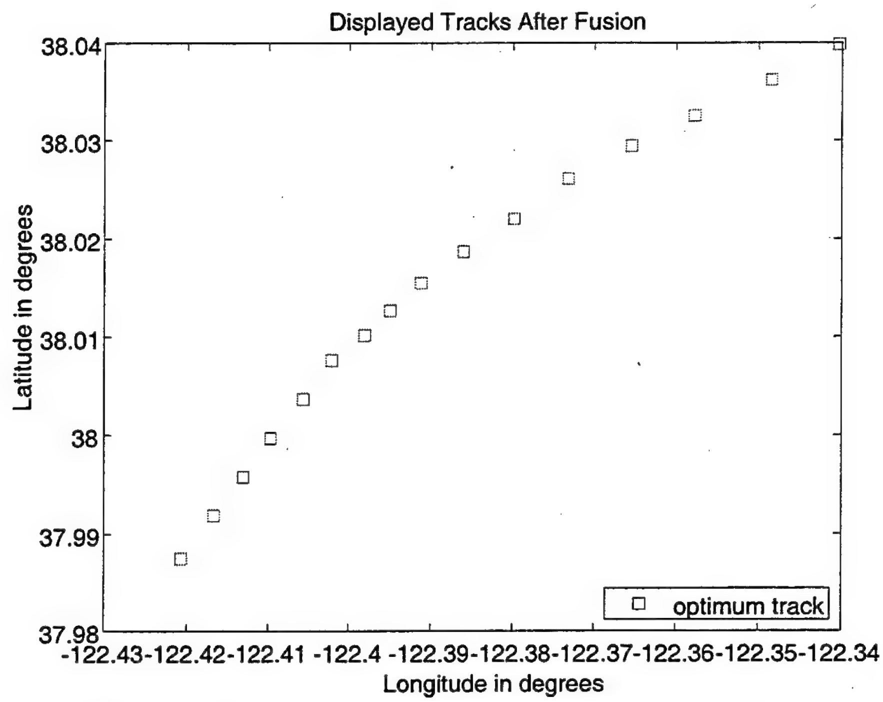


Figure 6.8.b. VTS San Francisco Track SAN JOAQUIN, Two Sensors, Two Tracks, Two Attributes (latitude and longitude): fusion applied

The next scenario involved the vessel MARIN TWILIGHT on a northbound course transiting through the same area. The vessel's path history was initially tracked by the Point San Pablo site (Sensor 2), and recorded observations commenced once the Mare Island site (Sensor 1) detected the vessel and displayed a duplicate track. Figure 6.9 shows the duplicate tracks reported by the two sensors.

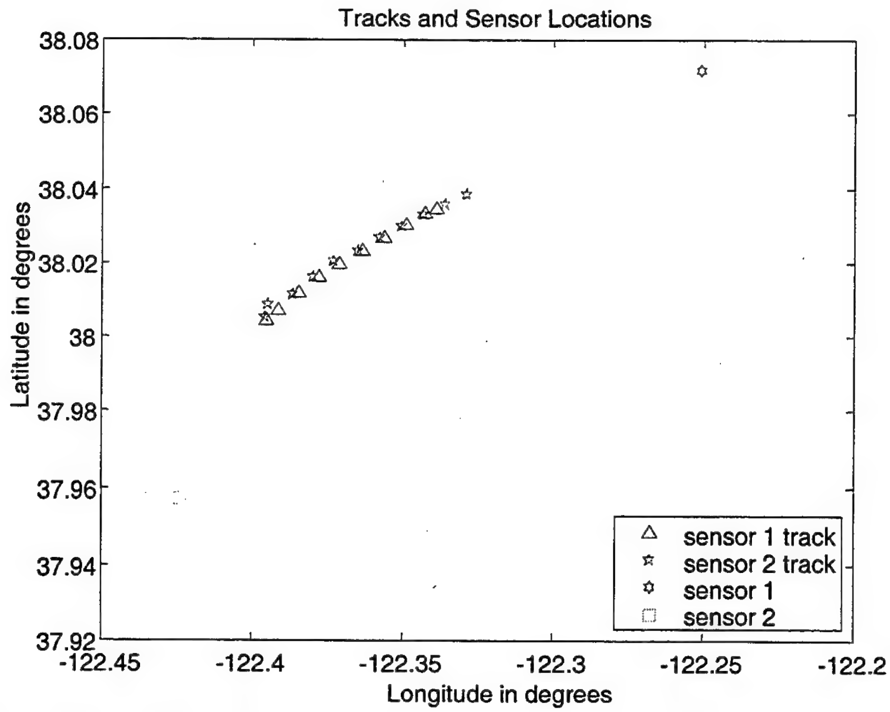
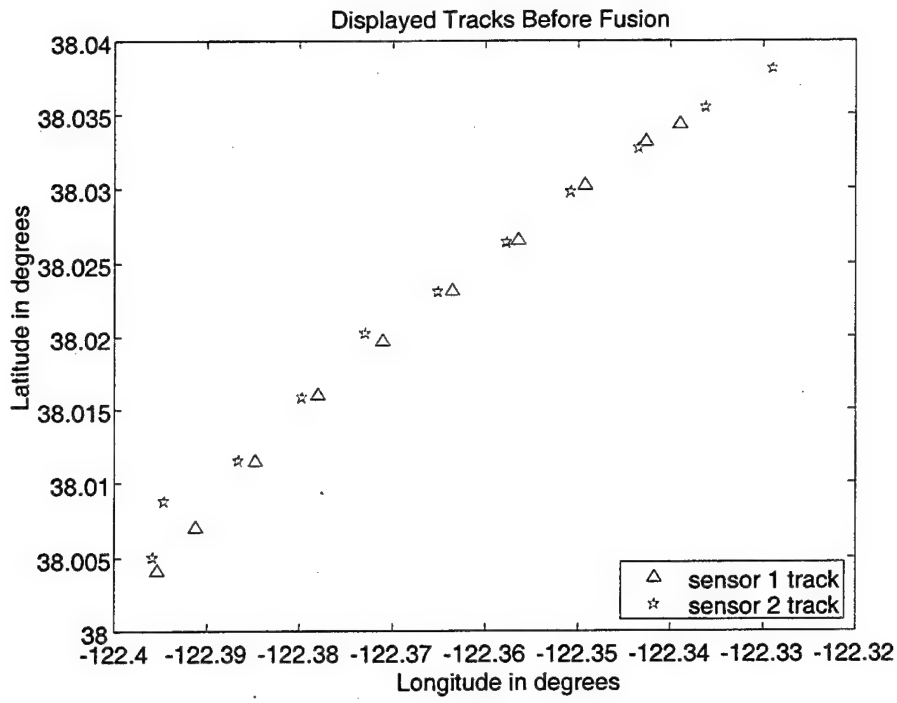
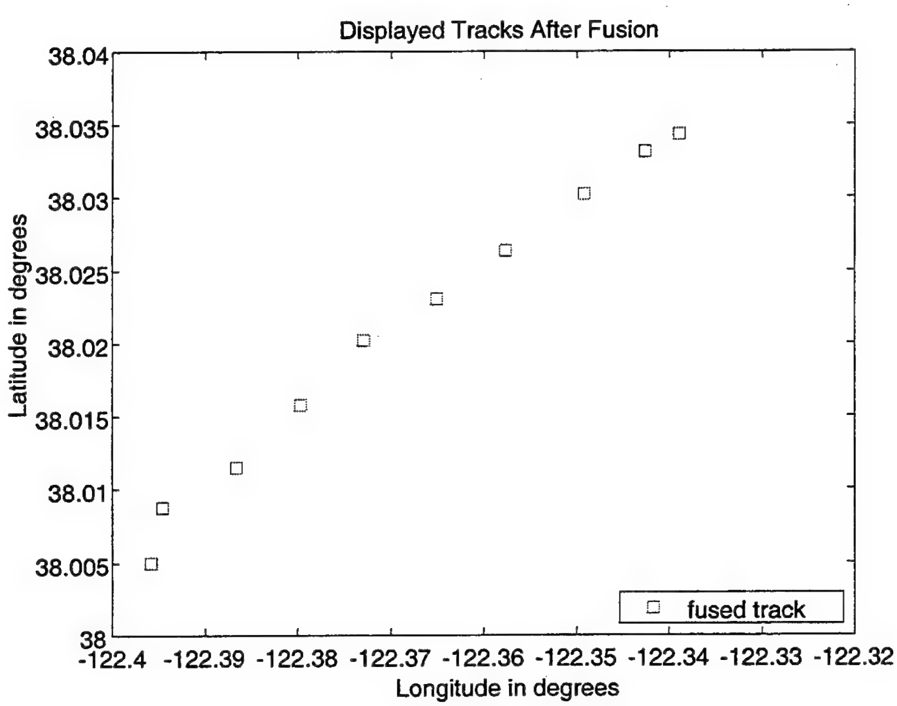


Figure 6.9. VTS San Francisco Track MARIN TWILIGHT with Sensor Sites: No Fusion Applied

Similar to the SAN JOAQUIN track data, the longitudinal separation between the MARIN TWILIGHT tracks was approximately 0.096 NM or 194 yards. Figure 6.10.a shows the duplicate tracks before the fusion process. After applying the algorithm to the data set, the tracks were fused, and the closest sensor site, Point San Pablo, was selected. Sensor swapping occurred at approximately 5.3 NM from the Point San Pablo site and 5.6 NM from the Mare Island site. Sensor 2 was selected to track MARIN TWILIGHT for the remaining scans. Figure 6.10.b shows the fused track displayed to the VTS operator.



(a)



(b)

Figure 6.10. VTS San Francisco Track MARIN TWILIGHT, Two Sensors, Two Tracks, Two Attributes (latitude and longitude): (a) no fusion applied and (b) fusion applied

Other similar scenarios from VTS San Francisco were tested with identical results. The duplicate tracks were fused, and the sensor swap occurred at approximate distances of 5.2 to 5.5 NM away from either sensor site. Range from the target to the sensor site contributed to the selection of the optimum sensor. Because the data were localized in a particularly narrow traffic channel, the range at which the optimum sensor was swapped was nearly the same distance for all scenarios in this region. However, the scenarios proved useful in demonstrating the effectiveness and adaptability of the fusion algorithm.

VII. CONCLUSION

This thesis focused on a fuzzy association based fusion algorithm to minimize duplicate target tracks caused by overlapping sensor coverage, a problem inherent to multisensor/multitarget (MSMT) tracking systems. The U.S. Coast Guard Vessel Traffic Service System (VTS) is a MSMT system, where duplicate target tracks are displayed to the VTS operator in regions of sensor overlap, such as Puget Sound and San Pablo Bay. To improve the effectiveness of the VTS operator to monitor and manage vessel traffic, a data fusion algorithm based on fuzzy association was proposed to fuse redundant tracks and automate the track correlation and sensor selection processes that are currently performed by VTS operators manually.

A. DISCUSSION OF RESULTS

The proposed fusion algorithm was based on a centralized positional fusion model using a fuzzy associative technique that uses the Fuzzy Clustering Means (FCM) algorithm. The fusion algorithm was applied to "real-life" vessel traffic scenarios from USCG VTS systems and proved effective in fusing duplicate tracks. The algorithm has several advantages. It required only the present data and sensor resolutions to fuse tracks. It did not require data from previous scans to fuse redundant tracks. The algorithm also handled varying numbers of measurements, sensors, and attributes, which made it applicable to sparse as well as dense traffic environments. Membership values were computed from present data and the resolutions of the sensors, which resulted in data adaptive calculation for each sensor scan. The algorithm also required considerably fewer IF-THEN rules since measurement-to-measurement association was performed by searching for the highest membership value in the partition matrix, rather than applying linguistic variables and predetermined thresholds for associating each measurement. The fusion of duplicate tracks was accomplished by selecting the most accurate sensor, eliminating the need for computing a composite or fused estimate. In the scenarios tested, the most accurate sensor was selected, and duplicate tracks were reduced.

B. SUGGESTIONS FOR FURTHER STUDY

The findings in this study provide a solution to the overlapping sensor coverage problem and are not entirely conclusive. Further studies need to be done to test and enhance the performance of the proposed algorithm.

1. Additional Attributes

The data collected from VTS San Francisco were limited to the latitude and longitude attributes. Additional attributes such as course and speed can be obtained from the VTS and integrated into the fusion algorithm. Adding attributes to the fusion algorithm has proven to be non-detrimental to the performance of the algorithm and easily incorporated by the FCM algorithm. In addition, sensor resolutions would have to be determined for each these new attributes.

2. Complex Scenarios

The scenarios presented in this study were typical traffic patterns encountered at VTS systems. The two track and two sensor scenario is the most common redundant target situation VTS operators experience. Further testing of the algorithm in more complex scenarios is necessary to prove its effectiveness and accuracy in correlating the multiple tracks to multiple sensors. Complex data sets such as overlapping sensor coverage of multiple vessels in crossing or overtaking situations or break-offs of vessels-in-tow provide a challenge to the fusion algorithm. Data sets can be computer generated for simulation or collected from different VTS systems. The latter set would provide more realistic results; however, coordination among several VTS systems would be required in order to observe and record these rare but complex situations.

3. Verification of the Proposed Algorithm

The proposed algorithm is able to correlate and fuse multiple tracks with "real-life" traffic data in simulated real-time. A step in advancing this algorithm is to implement it at an operational VTC, a system with real-time track measurements, and test its efficiency in

correlating and fusing redundant tracks on site. Since the algorithm resides between the track database manager (Tbdm) and the operator's display console, a stand-alone PC running the algorithm may process sensor information in parallel with the database and operator display processors in a non-intrusive manner. Further work would be required to encode the algorithm into the current VTS software.

APPENDIX A. VTS SAN FRANCISCO TRACK STATUS CODES

The following codes are used at VTS San Francisco to identify the status of observed tracks:

DT	Drop tow
PT	Pickup two
TR	Track Start Radar
TS	Track Start Standard Route
RL	Radar Lost
RC	Radar Coast
RR	Radar to Radar Transfer
SR	Standard Route to Radar
RS	Radar to Standard Route
AL	Alarms
OU	Operator Update
H	Handoff
SU	System Update
SS	Standard Route to Standard Route
LS	Lost to Standard Route
L	Lost Track
C	Correlated Track
DC	Decorrelated Track
F	Fuse
DF	De-Fuse
TM	Track Start Manual

AM	AIS to Manual
AS	AIS to Standard Route
MA	Manual to AIS
MR	Manual to Radar
MS	Manual to Standard Route
RM	Radar to Manual Transfer
SA	Standard Route to AIS
SM	Standard Route to Manual
TF	Track Start Fused
TC	Track Start Correlated
TA	Track Start AIS

APPENDIX B. DATA CAPTURE ALGORITHM

```
%DATA CAPTURE ALGORITHM
%getdatax.m
%THIS FUNCTION TAKES IN DATA SUPPLIED BY THE USCG AND PUTS IT IN A
%FORMAT THAT CAN BE USED BY THE FUSION ALGORITHM.  THE OUTPUT IS
AN %OBSERVATION MATRIX WHICH SIMULATES THE TDBM.


BVesselName = '                          ';    % 26 spaces for padding

%BTrackStatus = '     ';% 5 spaces for padding

% Initialize Storage vectors

VesselName = [];
UTC = [];
TrackStatus = [];
TrackIDNumber = [];
SensorTrackNumber = [];
TrueCourse = [];
Speed = [];
Latitude = [];
Longitude = [];
Size = [];
TrackQuality = [];

filename = input('Enter file name » ','s');

% ---------------Start reading the file -------------

fid = fopen(filename,'r');          % Read only

st = fgets(fid);                    % Get first line

while st ~= [ -1 ];                 % Check for end-of-file N = 1:10

Cloc = findstr(st,',');             % Finds delimiter

VesselName= [VesselName; st(1:Cloc(1)-1),BVesselName(1:26-length(st(1:Cloc(1)-1)))];

UTC = [UTC; str2num(st(Cloc(1)+1:Cloc(2)-1))];
```

```
TrackStatus = [TrackStatus; str2num(st(Cloc(2)+1:Cloc(3)-1))];

%BTrackStatus(1:5-length(st(Cloc(2)+1:Cloc(3)-1)))];

TrackIDNumber = [TrackIDNumber; str2num(st(Cloc(3)+1:Cloc(4)-1))];

SensorTrackNumber =[SensorTrackNumber;str2num(st(Cloc(4)+1:Cloc(5)-1))];

TrueCourse = [TrueCourse; str2num(st(Cloc(5)+1:Cloc(6)-1))];

Speed = [Speed; str2num(st(Cloc(6)+1:Cloc(7)-1))];

Latitude = [Latitude; str2num(st(Cloc(7)+1:Cloc(8)-1))];

Longitude = [Longitude; str2num(st(Cloc(8)+1:Cloc(9)-1))];

Size = [Size; str2num(st(Cloc(9)+1:Cloc(10)-1))];

TrackQuality = [TrackQuality; str2num(st(Cloc(10)+1:Cloc(10)+2))];

st = fgets(fid);
end

% BUILD DATA BASE OF ALL OBSERVATIONS PRESENT IN TDBM

ObsnMatrix=[Latitude Longitude TrueCourse Speed Size TrackIDNumber UTC
TrackQuality TrackStatus SensorTrackNumber];
```

APPENDIX C. FUZZY CLUSTERING MEANS ALGORITHM CODE

```
%%%%%%%%%%%%%%%%%%%%%%
%%%         vts01.m            %%%
%%%      Tracks 830 & 831       %%%   (830831test.dat)
%%%   Puget Sound 1996 Data    %%%
%%%       FCM Algorithm         %%%
%%%%%%%%%%%%%%%%%%%%%%

%%% Created by: LTJG E.S. Anzano and Major Ashraf M. Aziz  %%%
%%%                        May 1999                        %%%

%%% Algorithm Developed By: Major Ashraf M. Aziz     %%%

%% getdatax function gets ASCII dataset

clear all
getdatax     %% "830831test.dat"

obs=ObsnMatrix;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

nsamples=96;
c1=0;  % counter for correct correlation
for kk=1:nsamples     %  kk the index of the sample (TIME or nsamples)
kk1=kk;
kk2=kk1+109;
zobsx1=obs(kk1,1)*1e-2;
zobsx2=obs(kk2,1)*1e-2;
zobsy1=obs(kk1,2)*1e-2;
zobsy2=obs(kk2,2)*1e-2;
zobsz1=obs(kk1,3);
zobsz2=obs(kk2,3);

x1minutes = (zobsx1-47)*1e2/60;              %Conversion of minutes to degrees
x2minutes = (zobsx2-47)*1e2/60;
y1minutes = abs((zobsy1+122)*1e2/60);
y2minutes = abs((zobsy2+122)*1e2/60);

zobsx1 = 47 + x1minutes;              %Adding converted minutes to degrees
zobsx2 = 47 + x2minutes;
zobsy1 = -(122 + y1minutes);
zobsy2 = -(122 + y2minutes);
```

```
%%% Calculate distance vector from Radar station to target

R = 3443.9; %%% radius of Earth in Nautical Miles

%%% West Point Radar
WPobs = [47.66222; -122.432777];

phi1 = WPobs(1,1)*pi/180;  %%convert to radians
phi2 = zobsx1*pi/180;
theta1 = WPobs(2,1)*pi/180;
theta2 = zobsy1*pi/180;

AOB1 = acos(cos(phi1)*cos(phi2)*cos(theta2-theta1)+sin(phi1)*sin(phi2));
D1 = R*AOB1;  %% distance from West Point Radar to Target in Nautical Miles

%%% Pier 36 Radar

P36Pobs = [47.59; -122.3494445];

phi3 = P36Pobs(1,1)*pi/180;
phi4 = zobsx2*pi/180;
theta3 = P36Pobs(2,1)*pi/180;
theta4 = zobsy2*pi/180;

AOB2 = acos(cos(phi3)*cos(phi4)*cos(theta4-theta3)+sin(phi3)*sin(phi4));
D2 = R*AOB2;  %% distance from P36 Radar to Target in Nautical Miles

distance = [D1 D2; D1 D2]                    % Range from targets to sensors
zobs(:,1)=[zobsx1;zobsy1;zobsz1];
zobs(:,2)=[zobsx2;zobsy2;zobsz2];

track1x(kk)=zobsx1;
track1y(kk)=zobsy1;
track2x(kk)=zobsx2;
track2y(kk)=zobsy2;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

n=2;      %number of reported tracks
m=3;      % number of attributes
mm=1;     % factor of the resolution (from 1 to 3)

rres = 0.00375*abs(distance).*[1/60 1.25/60; 1/60 1.25/60];  %Resolution Lat/Long degree
bres = [0.35; 0.35]                                          %Bearing resolution
res = [rres bres];                                           %in degrees decimal
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Similarity Measures and Sensor Error Calculations
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for i=1:n
for j=1:n
if j==i
dd=res(:,i);
else
dd=zobs(:,i)-zobs(:,j);
end

d(i,j)=sqrt(dd'*dd);          % d will be a symmetric matrix
end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Membership Value Calculation
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% m is the deffuzification constant, 1 < m <infinity

m=3;
for i=1:n
for k=1:n
cc=0;
for j=1:n
temp1=(d(i,k)/d(j,k))^(2/(m-1));
cc=temp1+cc;
end
ures(i,k)=1/cc;
end
end

%ures
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Finding the optimum sensor   (maximum grade, i.e. high resolution)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for i=1:n
ss(i)=ures(i,i);
end

[Y,I]=sort(ss);     %ascending order

for i=1:n
  opts(i)=I(n-i+1);
```

```
end

%opts          % gives the priority when two tracks are same

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Comparison with horizontal thresholds
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for i=1:n
for j=1:n
udiff(i,j)=(ures(i,j)-ures(i,i));
if udiff(i,j)>=0
u(i,j)=1;
else
u(i,j)=0;
end

end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Comparison with vertical thresholds
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for i=1:n
for j=1:n
udiff(j,i)=(ures(j,i)-ures(i,i));
if udiff(j,i)>=0
uv(j,i)=1;
else
uv(j,i)=0;
end

end
end

%uv
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

uf=zeros([n,n]);
for i=1:n
for j=1:n

if u(i,j)==1
if u(j,i)==1
if uv(i,j)==1
```

```
if uv(j,i)==1
uf(i,j)=1;
uf(j,i)=1;
end
end
end
end

end
end

%%%%Put zeros in the diagonal elements%%%%%%%%%%%%%%

for i=1:n
uf(i,i)=0;
end
decision=uf ;        % final decision
%pause

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Calculation of the number of the correct removing the
% redundant tracks
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

for i=1:n
zx=0;
for j=1:n
if j==i
%nothing to do
else
zx=zx+decision(i,j);
end            % if
end            % for j

if i==1
if zx==1
c1=c1+1;
end
end
end          %for i

%pause
%end      % END FOR KK
```

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Calculation of the percentage of the correct removing the redundant tracks
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

pc1=(c1/nsamples)*100;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Determination of the optimum track
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

if opts(1)==1
        trackxf(kk)=track1x(kk);
        trackyf(kk)=track1y(kk);
        optimumtrack = 'sensor 1'
elseif opts(1)~=1
        trackxf(kk)=track2x(kk);
        trackyf(kk)=track2y(kk);
        optimumtrack = 'sensor 2'
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Plotting the results
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% 1- Reported tracks
%%%%%%%%%%%%%%%%%%%%%%%%

figure(1)
plot(trackyf,trackxf,'rs');

 xlabel('Longitude ddd.dd'),ylabel('Latitude dd.dd')
 title('Fig. 2 Displayed Tracks After Fusion ');

ymax = ceil(10*max(obs(:,1)))/1e3;
ymin = floor(10*min(obs(:,1)))/1e3;
xmax = floor(10*min(obs(:,2)))/1e3;
xmin = ceil(10*max(obs(:,2)))/1e3;

ymaxminutes = (ymax-47)*1e2/60;
yminminutes = (ymin-47)*1e2/60;
xmaxminutes = abs((xmax+122)*1e2/60);
xminminutes = abs((xmin+122)*1e2/60);

ymax = 47 + ymaxminutes;
ymin = 47 + yminminutes;
xmax = -(122 + xmaxminutes);
xmin = -(122 + xminminutes);
```

```
axis([xmax xmin ymin ymax])
 legend('optimum track', 4)

 figure(2)
 plot(track1y,track1x,'^',...
   track2y,track2x,'p');

 xlabel('Longitude ddd.dd'),ylabel('Latitude dd.dd')
 title(' Fig. 1 Displayed Tracks Before Fusion ');
 axis([xmax xmin ymin ymax])
 legend('sensor 1 track', 'sensor 2 track', 4)

% 3- Final tracks
%%%%%%%%%%%%%%%%%%%%

figure(3),
plot(track1y,track1x,'^', ...
  track2y,track2x,'p',...
  trackyf,trackxf,'s',...
  WPobs(2,1),WPobs(1,1),'bh',...
  P36Pobs(2,1),P36Pobs(1,1),'gh');

title(' Fig. 3 Displayed Tracks After Fusion ');
xlabel('Longitude ddd.dd'),ylabel('Latitude dd.dd')
legend('sensor 1 track', 'sensor 2 track', 'optimum track', 'sensor 1', 'sensor 2' ,1)


end
```

LIST OF REFERENCES

1. Aziz, A., Tummala, M., Cristi, R., "Fuzzy Logic Data Association Approach In Multisensor-Multitarget Tracking Systems," *Proceedings of the IEEE IASTED International Conference on Signal and Image Processing*, Las Vegas, Nevada, October 1999.

2. Aziz, A., Tummala, M., Cristi, R., "Fuzzy Logic Data Correlation Approach in Multisensor-Multitarget Tracking Systems," Accepted For Publication on Signal Processing, EURSIP, *European International Journal*, Vol., No., July 1999.

3. Midwood, Sean A., "A Computational Efficient and Cost Effective Multisensor Data Fusion Algorithm for the U.S. Coast Guard VTS," Master's Thesis, Naval Postgraduate School, Monterey, California, September 1997.

4. Glenn, Ian N., "Multilevel Data Association for the VTS System and the Joint Maritime Command Information System," Master's Thesis, Naval Postgraduate School, Monterey, California, December 1995.

5. Ruthenberg, Thomas M., "Data Fusion Algorithm for the VTS System: A Fuzzy Associative System Approach," Master's Thesis, Naval Postgraduate School, Monterey, California, September 1994.

6. USCG VTS Puget Sound, Home Page, [http://www.uscg.mil/d13/units/vts/psvts.html]. May 1999.

7. USCG VTS San Francisco, Home Page, [http://www.uscg.mil/d11/vtssf/]. January 1999.

8. Hall, David L., *Mathematical Techniques in Multisensor Data Fusion*, Artech House Inc., Norwood, Massachusetts, 1992.

9. Aziz, A., Tummala, M., Cristi, R., "A Time Series of Decisions Approach in Detection Systems," Manuscript, Naval Postgraduate School, Monterey, California, 1998.

10. Waltz, E., Llinas, J., *Multi-Sensor Data Fusion*, Artech House, Inc., Boston, Massachusetts, 1990.

11. Kosko, Bart, *Neural Networks and Fuzzy Systems*, Prentice Hall Inc., Englewood Cliffs, New Jersey, 1992.

12. Lootsma, Freerk A., *Fuzzy Logic for Planning and Decision Making*, Kluwer Academic Publishers, Dordrecht, Netherlands, 1997.

13. Yoder, C.F., "Astrometric and Geodetic Properties of Earth and the Solar System," *Global Earth Physics, A Handbook of Physical Constants*, American Geophysical Union, 1995.

14. Math Forum, "Longitude and Latitude to Determine Distance," [http://mirrors.org.sg/mathforum/dr.math/problems/atlas5.21.96.html]. May 1996.

INITIAL DISTRIBUTION LIST

No. Copies

1. Defense Technical Information Center 2
   8725 John J. Kingman Rd., STE 0944
   Ft. Belvoir, VA 22060-6218

2. Dudley Knox Library 2
   Naval Postgraduate School
   411 Dyer Rd.
   Monterey, CA 93943-5101

3. Chairman, Code EC 1
   Department of Engineering & Technology
   Naval Postgraduate School
   Monterey, CA 93943-5121

4. Curriculum Officer, Code 34 1
   Engineering and Technology Curriculum
   Naval Postgraduate School
   Monterey, CA 93943-5109

5. Professor Murali Tummala, Code EC/Tu 2
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, CA 93943-5121

6. Professor Roberto Cristi, Code EC/Cx 2
   Department of Electrical and Computer Engineering
   Naval Postgraduate School
   Monterey, CA 93943-5121

7. Ashraf M. Aziz 1
   SMC 1800
   Naval Postgraduate School
   Monterey, CA 93943-1800

8. LCDR Dean Bruckner 1
   Department of Electrical Engineering
   U.S. Coast Guard Academy
   15 Mohegan Avenue
   New London, CT 06320

9. Commanding Officer..1
Command and Control Engineering Center
4000 Coast Guard Boulevard
Portsmouth, VA 23703

10. LTJG Eugenio Anzano...2
Electronic Systems Division
Building 54
Maintenance and Logistics Command Pacific
Coast Guard Island
Alameda, CA 94501